Universidade do Minho
Escola de Engenharia

Ana Margarida Cunhal Fontaínhas

Injection moulding simulation using OpenFOAM®

Dissertação de Mestrado

Mestrado Integrado em Engenharia de Polímeros

Trabalho efetuado sob a orientação do:

Doutor Célio Bruno Pinto Fernandes

Professor Doutor João Miguel Nóbrega

Janeiro 2019

DECLARAÇÃO

Nome: Ana Margarida Cunhal Fontaínhas Endereço eletrónico: anamargaridafontainhas@gmail.com
Telefone: 00351 913505748

Número do Bilhete de Identidade: 14789997

Título da dissertação: Injection moulding simulation using OpenFOAM®

Orientador(es): Doutor Célio Bruno Pinto Fernandes e Professor Doutor João Miguel Nóbrega

Ano de conclusão: 2019

Designação do Mestrado:

Ciclo de Estudos Conducente ao Grau de Mestre em Engenharia de Polímeros

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA DISSERTAÇÃO, APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE;

Universidade do Minho, ___/___/______

Assinatura: ____

"And will you succeed?
Yes! You will, indeed!
(98 and ¾ percent guaranteed.)

KID, YOU'LL MOVE MOUNTAINS!"

– Dr. Seuss

ACKNOWLEDGMENTS

A presente dissertação não teria sido possível sem o suporte de várias pessoas e entidades. Assim, eu gostaria de manifestar a minha gratidão a todos que direta ou indiretamente contribuíram para a conclusão deste trabalho.

Antes de mais, gostaria de agradecer aos meus orientadores, o Doutor Célio Bruno Pinto Fernandes e o Professor Doutor João Miguel Nóbrega, por toda a disponibilidade e paciência, pelo apoio e aconselhamentos prestados e por todo o encorajamento prestado ao longo de todo o projeto, sem eles não teria sido possível.

À Engenheira Marisa Mendes, por toda a partilha de conhecimento em relação ao programa Moldex3D® e por toda a disponibilidade prestada.

To Kristjan Krebelj, for being kind enough to have shared his knowledge about the solver he developed, for the availability and interest in this project.

The execution of most of the simulation was only possible due to the use of the computational cluster, so I would like to acknowledge the Search-ON2: Revitalization of HPC infrastructure of UMinho, (NORTE-07-0162-FEDER-000086), co-founded by the North Portugal Regional Operational Programme (ON.2-0 Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF).

Aos amigos de sempre e aos que fiz nestes 5 belos anos de universidade. Um obrigada especial à Cátia Araújo que desde o dia um nesta universidade me acompanha e que ao longo desta etapa não foi diferente, à Adriana Dias por transmitir sempre boa energia e sempre me apoiar, à Rita Leite por estar sempre presente e à Sónia Miranda, João Pinho, Mário Silva, Miguel Gonçalves, Leandro Fernandes, Carina Lopes e Eva Silva um grande abraço pelo companheirismo.

Por fim e porque foram e são o meu pilar mais importante, quero agradecer à minha família. Aos meus avós paternos que não conseguiram acompanhar em vida este culminar, mas que sempre contribuíram e acreditaram no meu sucesso. À minha avó Maria Teresa pela sua preocupação e pelo seu cuidado constante comigo. À minha estimada prima Ana Alexandra por me conhecer tão bem e saber sempre como me animar. Aos meus pais por terem sempre acreditado em mim e por serem o meu suporte. Com um carinho muito grande ao meu querido irmão, por ser desde sempre o meu exemplo a seguir, obrigada por todas as "injeções" de confiança que foram fundamentais para poder terminar esta etapa.

A todos o meu sincero obrigada! To all my most honest thank you!

ABSTRACT

Injection moulding simulation using OpenFOAM®

The injection moulding process is one of the most important technique for the production of plastic parts, in which competition has been pushing the industry to increase parts quality at the lowest possible cost. For that it is necessary to use simulation programs, but the dissemination of these tools by the industry faces two major difficulties: the cost of commercial software licenses and/or the lack of knowledge to use these tools properly. The cost limitation can be overcome by using open-source modelling software, such as the solvers implemented in OpenFOAM® computational library.

The motivation of this work is then focused on validating a solver developed for simulating the injection moulding process in OpenFOAM®. Simulations were performed in openInjMoldSim solver and in the commercial software, widely used by industry, Moldex3D®. For the simulations, three meshes were generated starting with a coarse mesh and ending with a more refined mesh. After the meshes were generated, the same boundary and initial conditions were defined in both programs. To validate the solver, the results of the melt flow front, temperature, velocity and pressure obtained by the simulations were analysed and compared.

With the realization of this project, it was possible to create a methodology to compare the commercial software, Moldex3D®, with the openInjMoldSim solver. Through the analysis and comparisons made, the results of openInjMoldSim indicated that they converged to the results of commercial software. However, in order to be sure that the results converged, it is necessary to further refine the meshes until they are converged. With the obtained results it was not possible to validate the solver openInjMoldSim, but the accomplishment of this work allowed to be one step closer to the main objective.

Keywords

Injection moulding process, numerical modelling, openInjMoldSim, Moldex3D®

RESUMO

Simulação do processo de injeção utilizando OpenFOAM®

O processo de moldação por injeção é uma das técnicas mais importantes para a produção de peças plásticas, em que a concorrência tem vindo a obrigar a indústria a aumentar a qualidade das peças com o menor custo possível. Para isso é necessário recorrer a programas de simulação, mas a disseminação deste tipo de ferramentas enfrenta duas grandes dificuldades: o custo das licenças dos softwares comerciais e/ou a falta de conhecimento para utilizar estas ferramentas adequadamente. O custo das licenças pode ser ultrapassado utilizando softwares de modelação open-source, como é o caso de solvers implementados na biblioteca computacional OpenFOAM®.

A motivação deste trabalho focou-se na validação de um solver desenvolvido para a simulação do processo de injeção em OpenFOAM®. Para isso foram realizadas simulações no solver openInjMoldSim e no software comercial, muito utilizado pela indústria, Moldex3D®. Para a realização das simulações foram geradas três malhas, iniciando por uma malha mais grosseira e terminando com uma malha mais refinada. Depois de geradas as malhas foram definidas as mesmas condições de fronteira e iniciais em ambos os programas. Para validar o solver, os resultados da frente de fluxo, temperatura, velocidade e pressão obtidos pelas simulações foram analisados e comparados.

Com a realização deste projeto foi possível criar uma metodologia para comparar o software comercial com o solver openInjMoldSim. Pela análises e comparações realizadas os resultados do openInjMoldSim indicavam convergir para os resultados do software comercial. Contudo, para ter a certeza que os resultados convergem, é necessário refinar mais as malhas até obter malhas convergidas. Com os resultados obtidos ainda não foi possível validar o solver openInjMoldSim, mas a realização deste trabalho permitiu estar um passo mais próximo do objetivo principal.

Palavras-chave

Processo de moldação por injeção, modelação numérica, openInjMoldSim, Moldex3D®.

CONTENTS

Acknowledgments vii

Abstract ix

Resumo xi

Figures Index xv

Tables Index xix

Abbreviations and acronyms list xxi

Chapter I – Introduction 1

1. Motivation 1
2. Objectives 1
3. Thesis structure and Organization 2

Chapter II – Theoretical Introduction 3

1. Injection moulding 3
    - 1.1 Introduction 3
    - 1.2 Process description 3
2. Injection moulding materials 5
    - 2.1 Types of polymers 5
    - 2.2 Rheology 6
    - 2.3 Thermal properties 14
    - 2.4 Pressure-Volume-Temperature Behaviour 19
3. Governing equations for the injection moulding process 22
    - 3.1 Mass conservation equation 22
    - 3.2 Momentum conservation equation 22
    - 3.3 Energy conservation equation 23
4. Numerical methods 24
    - 4.1 Finite-difference method 25
    - 4.2 Finite-element method 25
    - 4.3 Boundary-element method 25
    - 4.4 Finite-volume method 26

Chapter III – State of the art 27
1. Introduction 27
2. Mathematical models used to describe the injection moulding process 27
    2.1 One dimensional approaches 27
    2.2 Two-dimensional approaches 31
    2.3 Three-Dimensional Formulation 35
3. Available software created for the injection moulding process 38

Chapter IV – Case Study 41
1. Introduction 41
2. Methodology 41
    2.1 Geometry 41
    2.2 Models used in the simulation 43
    2.3 Material 44
    2.4 Mesh generation 46
    2.5 Initial and boundary conditions 47
3. Results and Discussion 52
    3.1 Melt flow front 52
    3.2 Analysis of temperature predictions 58
    3.3 Analysis of velocity results 68
    3.4 Analysis of pressure results 77

Chapter V – Conclusions and Future work 83
1. Conclusions 83
2. Future work 85

References 87

FIGURES INDEX

Figure 1 – Injection moulding cycle. Adapted from [3] 4

Figure 2 – Representation of the fountain flow effect, the frozen layer and the heat loss to the mould. Adapted from [5]. 4

Figure 3 – The two plates model, where A is the area of the plate, H is the distance between plates, F is the force applied at the upper plate and **u** is the velocity of the moving plate. Adapted from [12]. .. 7

Figure 4 – Shear stress vs shear rate diagram for Newtonian and non-Newtonian fluids. Adapted from [14]. 8

Figure 5 – Characteristic viscosity curve of pseudoplastic fluids. Adapted from [7]. 9

Figure 6 – Molecular alignment with viscosity variations, adapted from [7]. 11

Figure 7 – Viscosity curve (red line) and approximation by the Power Law model (dashed line). Adapted from [17]. 11

Figure 8 – Viscosity approximation using the Bird-Carreau model. Adapted from [17]. 12

Figure 9 – Viscosity approximation with the Cross-WLF model. Adapted from [17]. 14

Figure 10 – Illustration of PVT-diagram for semicrystalline a) and amorphous polymers b). The specific volume is plotted as a function of the temperature and for different pressures. Adapted from [27]. 20

Figure 11 – Schematic diagram of a typical flow described by the Hele-Shaw model. Adapted from [4]. 31

Figure 12 – Geometry and dimensions of the part under study. 42

Figure 13 – Case study boundary surface groups. 42

Figure 14 – Phase indicator α distribution for a typical injection moulding case study. 43

Figure 15 – Mesh 1 from Moldex3D® and openInjMoldSim. 47

Figure 16 – Flow rate vs time obtained from the simulation result in Mesh 1 using Moldex3D®. ... 48

Figure 17 – Initial and boundary temperature conditions. 49

Figure 18 – Initial and boundary velocity conditions. 50

Figure 19 – Initial and boundary pressure conditions. 50

Figure 20 – Initial and boundary shear rate conditions. 51

Figure 21 – Initial and boundary concentration function conditions. 51

Figure 22 – Tests made for the amount of the initial filled cavity, in the case of Mesh 1. The polymer and air inside the cavity are represented by red and blue, respectively. 53

Figure 23 – Evolution of the flow rate over time for the different attempts of initial quantity of material inside the cavity in the simulations of the openInjMoldSim and for Moldex3D®. 54

Figure 24 – Evolution of flow rate over time for Mesh 2 for openInjMoldSim and Moldex3D simulations. 54

Figure 25 – Evolution of flow rate over time for Mesh 3 for openInjMoldSim and Moldex3D simulations. 55

Figure 26 – Melt flow front evolution for Mesh 1 at 0,036s, 0,038s, 0,040s and 0,042s of filling time, for both programs. 55

Figure 27 – Melt flow front evolution for Mesh 2 at 0,036s, 0,038s, 0,040s and 0,042s of filling time, for both programs. 56

Figure 28 – Melt flow front evolution for Mesh 3 at 0,036s, 0,038s, 0,040s and 0,042s of filling time, for both programs. 56

Figure 29 – Welding line position and temperature at which the melt flow front meet in Moldex3D®. 57

Figure 30 – Position of the line where the results were analysed. 58

Figure 31 – Temperature distribution of openInjMoldSim and Moldex3D® program at 0,076s of filling time for the Mesh 1. 59

Figure 32 – Temperature predictions by openInjMoldSim and Moldex3D® programs at 0,076s of filling time for the Mesh 1, with the new heat transfer coefficient condition in Moldex3D®. 60

Figure 33 – Points taken along the line where the results were analysed for Mesh 1 in Moldex3D®. 60

Figure 34 – Temperature predictions by openInjMoldSim and Moldex3D® program at 0,076s of filling time for the Mesh 2. 61

Figure 35 – Temperature predictions by openInjMoldSim and Moldex3D® program at 0,076s of filling time for the Mesh 3. 62

Figure 36 – Cross-section locations for which the results were analysed. 62

Figure 37 –Temperature distribution for Mesh 1 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs. 63

Figure 38 –Temperature distribution for Mesh 2 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs. 63

Figure 39 –Temperature distribution for Mesh 3 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs. 64

Figure 40 – Position of cross-section and line where the results were compared for Moldex3D®... 64

Figure 41 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 1.. 65

Figure 42 –Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 1. 65

Figure 43 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 2.. 66

Figure 44 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 2. 66

Figure 45 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 3.. 67

Figure 46 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 3. 67

Figure 47 – Flow direction velocity component distribution given by openInjMoldSim and Moldex3D®, at 0,076s of filling time for the Mesh 1. 69

Figure 48 – Flow direction velocity component distribution given by openInjMoldSim and Moldex3D®, at 0,076s of filling time for the Mesh 2. 69

Figure 49 – Flow direction velocity component distribution given by openInjMoldSim and Moldex3D®, at 0,076s of filling time for the Mesh 3. 70

Figure 50 – Comparison between temperature distribution in cross-section x=0,01m and x-velocity results for Moldex3D at 0,076s of filling time for the Mesh 3. 71

Figure 51 – Velocity contours for Mesh 1 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.. 72

Figure 52 – Velocity contours for Mesh 2 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.. 73

Figure 53 – Velocity contours for Mesh 3 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.. 73

Figure 54 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 1.. 74

Figure 55 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 1. 74

Figure 56 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 2. 75

Figure 57 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 2. 75

Figure 58 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 3. 76

Figure 59 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 3. 76

Figure 60 – Pressure contours for Mesh 1 at the end of filling time. 78

Figure 61 – Pressure drop variation for Mesh 1 in both programs. 78

Figure 62 – Pressure contours for Mesh 2 at the end of filling time. 79

Figure 63 – Pressure drop variation for Mesh 2 in both programs. 79

Figure 64 – Pressure contours of Mesh 3 at the end of filling time. 80

Figure 65 – Pressure drop variation for Mesh 3 in both programs. 80

TABLES INDEX

Table 1 – Melt temperature values for some well-known semicrystalline polymers [10]. 15

Table 2 – Glass transition temperature values for some well-known semicrystalline and amorphous polymers [10]. 15

Table 3 – Thermal conductivity values for some well-known semicrystalline and amorphous polymers [22]. 16

Table 4 – Specific heat capacity for some well-known semicrystalline and amorphous polymers *[10]*. 18

Table 5 – Thermo-physical properties and recommended conditions to process GPPS, based on MoldFlow® database. 45

Table 6 – Cross-WLF model parameters for GPPS – Styron 678 Americas Styrenics, based on MoldFlow® database. 45

Table 7 – Modified Tait-equation parameters for GPPS – Styron 678 Americas Styrenics, based on MoldFlow® database. 46

Table 8 – Number of cells in each mesh, in Moldex3D® and openInjMoldSim software. 47

Table 9 – Initial conditions used for the filling phase simulation with Moldex3D®......................... 48

Table 10 – Simulation time for each mesh in both programs. 52

Table 11 – Maximum x-velocities reached for the two meshes in both programs with the relative difference based on the average values. 77

Table 12 -Pressure drop for meshes 1, 2 and 3 in both programs with the relative difference based on the average values. 81

ABBREVIATIONS AND ACRONYMS LIST

®	Registered trademark
T	Temperature
T_g	Glass transition temperature
T_m	Melting temperature
A	Area
F	Force
$\boldsymbol{u}$	Velocity vector
H	Distance
τ	Shear stress
$\dot{\gamma}$	Shear rate
GNF	Generalized Newtonian fluids
$\boldsymbol{\tau}$	Shear stress tensor
$\dot{\boldsymbol{\gamma}}$	Shear rate tensor
μ	Dynamic viscosity
$\eta\left(\dot{\gamma}\right)$	Viscosity function
∇	Gradient vector
T	Transpose operation (superscript)
u	First component of velocity vector
v	Second component of velocity vector
w	Third component of velocity vector
η_0	Zero-shear rate viscosity
n	Power law index
η_∞	Viscosity at infinite shear rate
λ	Material time constant parameter for Bird-Carreau model
τ^*	Shear stress at the transition between Newtonian and power law behaviour
WLF	Williams-Landel-Ferry
PP	Polypropylene
HDPE	High-density polyethylene
PA	Polyamide

PET	Polyethylene terephthalate
ABS	Acrylonitrile butadiene styrene
PS	Polystyrene
PMMA	Poly(methyl methacrylate)
PC	Polycarbonate
PVC	Polyvinyl chloride
k	Thermal conductivity
c_p	Specific heat capacity
D	Thermal diffusivity
ρ	Density
PVT	Pressure Volume Temperature
$\hat{V}$	Material specific volume
LHS	Left hand side
$\boldsymbol{\sigma}$	Total stress tensor
$\mathbf{f}$	Body forces tensor
RHS	Right hand side
p	Pressure
P	Modified pressure
i	Internal energy
S_i	Source term
FDM	Finite-difference method
FEM	Finite-element method
BEM	Boundary element method
FVM	Finite-volume method
CFD	Computational fluid dynamics
CAE	Computer aided engineering
1D	1 Dimensional
3D	3 Dimensional
RFM	Radial flow model
FAN	Flow analysis network
BRFM	Bounded radial flow model

2D	2 Dimensional
CAD	Computer aided design
α	Concentration function
T_{melt}	Temperature of the melt to be injected
T_{mould}	Mould temperature
$T_{ejection}$	Ejection temperature
T_{freeze}	Temperature at which material solidifies
H	Heat transfer coefficient
GPPS	General purpose polystyrene
$\dot{\alpha}$	Instantaneous flow rate
$V_{fraction}$	Volume fraction
V_{part}	Volume of the part under study

CHAPTER I – INTRODUCTION

1. MOTIVATION

Nowadays, to optimize performance and minimize resources consumption, any project activities must be supported by appropriate modelling tools. In the market there are several modelling codes, capable of simulating extremely complex processes, such as Moldex3D® or MoldFlow® for the simulation of the injection moulding process. The dissemination of these codes by the industry faces two major difficulties: the cost of commercial software licenses and/or the lack of knowledge to use these tools properly. The cost limitation can be overcome by using open-source modelling software, such as the solvers implemented in OpenFOAM® computational library.

The motivation of this work is then focused on validating a solver developed for simulating the injection moulding process in OpenFOAM®, the openInjMoldSim, thus assessing it for application on the industrial side. For this purpose, a detailed comparison between the results obtained with openInjMoldSim and with a commercial software, widely used by the industry, Moldex3D®, will be performed.

2. OBJECTIVES

The main objective of this thesis is to compare the open-source solver openInjMoldSim with the commercial program Moldex3D®, in terms of accuracy. For that purpose, it is necessary to know the mathematical models that are used to model the injection moulding process and identify the ones that are employed in both programs. Additionally, for the purpose of comparing the results of the two programs it is imperative the same initial and boundary conditions, which are prescribed before the simulation starts, are equal or equivalent. Although this is a straightforward task in the open-source code, in a commercial code, in which some of the related decisions are not accessible to the user, it requires some investigation.

Once all the conditions have been defined and the simulation has been run in both programs the objective is to analyse the results and evaluate the validity of the open-source program when compared to the commercial one.

Throughout the work it is also an objective to identify limitations of both programs and, in the case of the open-source solver, to present suggestions for future improvements.

3. THESIS STRUCTURE AND ORGANIZATION

To describe the achievements concerning the proposed objectives and the results obtained in this research work, the present dissertation is subdivided into five chapters.

The present section, Chapter I, comprises the motivation and main objectives of the research work. Chapter II introduces the injection moulding process, the main governing equations considered for simulation purposes, the relevant physical properties of the materials employed in the process and the respective constitutive models considered required to simulate their behaviour, and the numerical methods employed in this application. In Chapter III the state of art is revised, by describing the evolution that has occurred over time regarding the numerical modelling of the injection moulding process, namely from 1D up to 3D and also shows available software created for injection moulding process. The case study used on the numerical validation of the open-source solver openInjMoldSim is presented and discussed in Chapter IV, comprising the pre-processing, results presentation and associated discussion, for the two employed modelling programs: openInjMoldSim and Moldex3D®. Lastly, Chapter V summarizes main results obtained and conclusions drawn from the work done. This last chapter comprises also recommendations for future work, which should allow the continuation and improvement of this research project.

CHAPTER II – THEORETICAL INTRODUCTION

1. INJECTION MOULDING

1.1 INTRODUCTION

Injection moulding is one of the prime processes for producing plastic parts, and it is estimated that 32 % of the plastic used in the manufacturing industry comes from injection moulding [1]. Many everyday items are made through injection moulding, such as mobile phone housings, automobile parts and lunch boxes [2]. This technique offers several advantages such as good surface finish, the capacity to process complex-shaped plastic parts, dispense with secondary operations, very high production rates, and low cost for mass production [3]. The following subsections describe the injection moulding in terms of process characteristics and mathematical governing equations.

1.2 PROCESS DESCRIPTION

The injection moulding is a cyclic process where moulds are used to shape the molten material that was forced to enter into the mould cavity. In a subsequent phase, the material cools down and solidifies inside the mould, obtaining a part that replicates the mould cavity geometry.

The injection moulding cycle is illustrated in Figure 1, where it begins when the mould closes (Figure 1.**1.**), and then the screw moves forward (Figure 1.**2.**) and forces the molten material to enter the mould (Figure 1.**3.**) – during the filling phase [4]. This phase is characterized by high flow rates, in which the dominant heat transfer mechanism is advection. However, mainly due to the high shear rates involved, heat is also generated by viscous dissipation, which depends on both deformation rate and material viscosity [2].

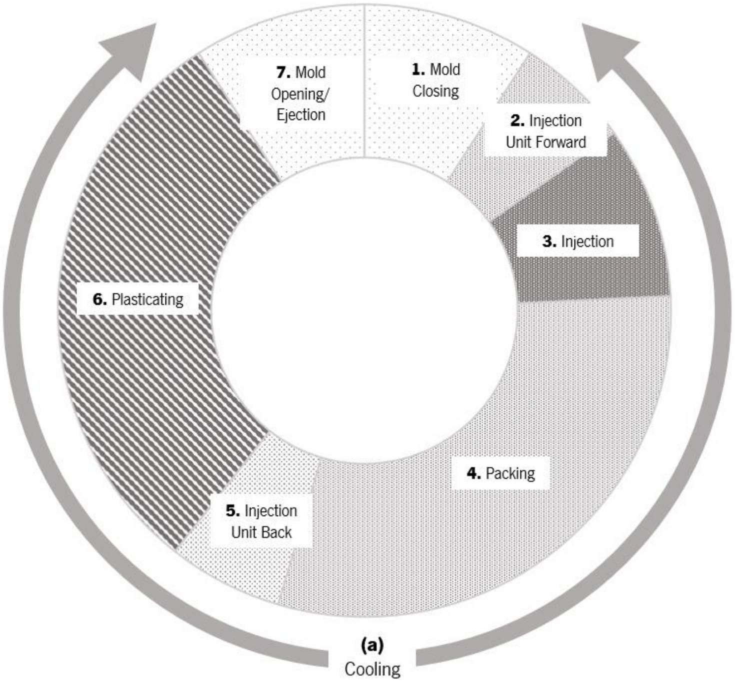

Figure 1 – Injection moulding cycle. Adapted from [3]

To obtain the final part, the mould, which is kept at a low temperature, cools down and solidifies the material. During this phase, the heat is removed by conduction through the mould wall and, subsequently, transferred to the cooling system. During injection as the material gets in contact with mould wall it cools down, and a thin layer of solidified material is formed. When more material is injected the existing one is forced to flow and, consequently, will extend the solidified layer along the entire mould wall. Along this phase the flow typical front pattern is called fountain flow and is illustrated in Figure 2. This "frozen layer" may rapidly reach an equilibrium thickness or continue to grow thereby restricting the flow of the incoming melt, which has a significant influence on the pressure required to fill the cavity, and an important role in the promoted shrinkage and warpage [2, 5].

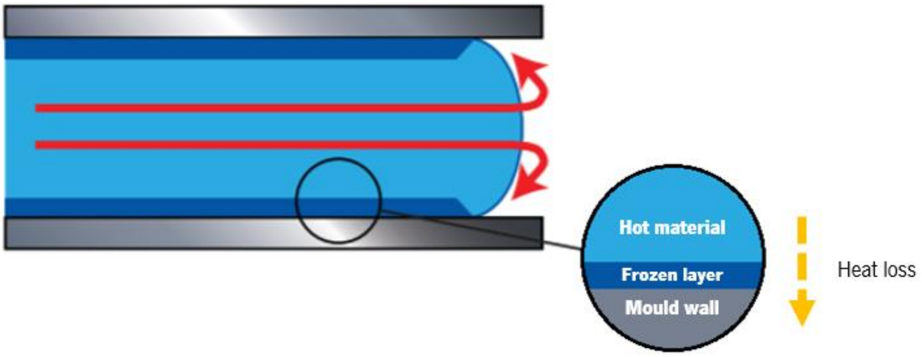

Figure 2 – Representation of the fountain flow effect, the frozen layer and the heat loss to the mould. Adapted from [5].

As soon as the cavity is filled begins the packing phase (Figure 1.**4.**). In this phase, an high pressure is maintained at the feed system inlet, allowing more material to enter the mould in order to compensate for volume reduction promoted by shrinkage [4]. From (Figure 1.**(a)**) it can be seen that the cooling phase dominates the process. That additional flow, due to material shrinkage is very small, thus both convection and viscous dissipation can be neglected. In this phase, the major heat transfer mechanism becomes the conduction. The frozen layer thickness continues to increase, and at some time the melt entrance freezes, thus the mould inlet pressure becomes ineffective and can be reduced [2].

In the next phase, the screw turns, plasticising the polymer melt for the subsequent shot to the front of the screw (Figure 1.**5.** and Figure 1.**6.**). During this phase the screw moves back, along its axis, to create a polymer melt reservoir of material at the end of the barrel, and simultaneously the part inside the mould is cooling. After reaching enough rigidity, the mould opens and the part is ejected (Figure 1.**7.**) [4].

Based on the description it is clear that the injection moulding process is rather complex, since it is a transient process that involves several heat transfer mechanisms, in conjunction with flow, and a phase change. To obtain a part with quality at the lowest possible cost it is necessary to understand the relationship between the geometry of the mould, material properties and processing conditions. This complex framework motivated the development of software to model the process for support design activities [3].

As described above during the injection moulding process the material deals with different temperatures, shear rates and pressures, therefore it is mandatory to be able to replicate numerically this behaviour using constitutive relationships that account for all the relationships between these parameters. The next section presents the most common constitutive equations generally used to describe the materials employed for the production of injected moulded parts.

2. INJECTION MOULDING MATERIALS

2.1 TYPES OF POLYMERS

There are two major classifications for synthetic polymers: thermoplastic and thermosets. Thermoset materials are cured in an irreversible way. The curing, which usually occurs rapidly under heat or ultraviolet light leads to the formation of a rigid three-dimensional lattice structure. After curing, the material will not remelt on heating [2, 6–8].

Thermoplastics are different because the molecules are not chemically joined. When heated, thermoplastics soften to a mobile flowable state where they can be shaped. Upon cooling, thermoplastics solidify and hold their shape. This process can be repeated for a number of times, provided the material does not undergo any permanent chemical change, as happens in thermal degradation [2, 6–8].

Thermoplastics and thermosets can be used in injection moulding, apart from the feeding system, the main difference in processing is the mould temperature. For the thermosets the mould is "hot" and for the thermoplastics the mould is "cold"[2]. Thermoplastics will be the principal focus of this project, since the reference open-source program only considers constitutive models appropriate for this class of material.

Thermoplastics can be classified as amorphous or semicrystalline. The molecules of amorphous polymers, at melt state, have mobility and adopt an entangled disordered configuration. When the polymer cools, it changes from rubbery to a transparent hard glass-like appearance. The temperature which this transition occurs is called the glass transition temperature, T_g [2, 6–8]. In the case of semicrystalline polymers, they possess crystalline regions, where the molecules are ordered, and amorphous regions. Semicrystalline polymers also present a glass transition temperature, T_g, but they will flow only when the temperature is above the crystalline melting point, T_m. The semicrystalline polymers are opaquer, because the crystallites scatter the light [6].

To select proper material, one must be acquainted of their rheological, thermal and mechanical properties [7], which will be described in the following three sub-sections.

2.2 RHEOLOGY

Rheology is the field of science that studies the deformation and flow of matter under forcing fields [9]. Among the variety of materials that rheologists study, polymers have been found to be the most interesting and complex [10].

The rheological properties have an important role in the injection moulding process. In mould filling, it is viscosity, along with thermal properties, that determines the pressure required to fill the mould cavity. For this reason, it is important to know in detail the rheological behaviour of polymers [7].

Therefore, it is necessary to define some fundamental rheological concepts, and for that the two-plates model, shown in Figure 3 is used. Consider a Couette flow experiment, where the upper plate with area A is set in motion by the force F and the resulting velocity $\boldsymbol{u}$ is measured, the lower plate is stationary (***u***=0 m/s), the distance between the plates is *H*, and the material sample is sheared in this shear gap [11].

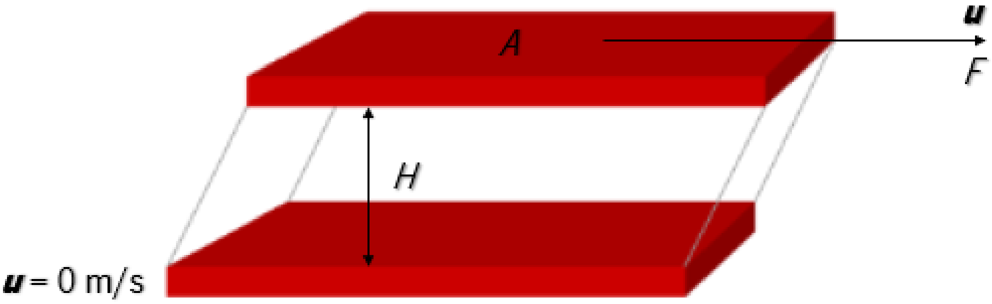

Figure 3 – The two plates model, where A is the area of the plate, H is the distance between plates, F is the force applied at the upper plate and ***u*** *is the velocity of the moving plate. Adapted from* [12].

When the upper plate drifts aside, the fluid is subject to a stress, the shear stress which is defined by

$$\tau = \frac{F}{A} \tag{1}$$

where *F* is the force and *A* is the area (see Figure 3). Another parameter that can be calculated for the Couette flow experiment is the shear rate, which is given by

$$\dot{\gamma} = \frac{\mathbf{u}}{H} \tag{2}$$

The shear rate provides information about the velocity gradient imposed on the flowing fluid. Viscosity is given as the ratio between shear stress and shear rate.

Now that the meaning of shear rate, shear stress and their relationship to viscosity is known, it is important to know that fluids can be classified as Newtonian or non-Newtonian. The main difference remains that Newtonian fluids exhibit a constant viscosity at all shear rates, while a non-Newtonian fluid is a fluid for which the ratio between the shear stress and the shear rate is not constant.

In practice, the ratio between shear stress and shear rate for Newtonian fluids is a straight line and the viscosity of the fluid remains constant as the shear rate is changed. In Figure 4 this relationship is presented in a diagram showing the behaviour of Newtonian and non-Newtonian fluids. In the case of the latter when the shear rate is changed, the shear stress does not vary in the same proportion. This phenomenon is caused by a modification of the viscosity, a sign of non-Newtonian fluids behaviour [13].

These are usually grouped in the following three categories [14]:

a) Time-independent fluids or generalized Newtonian fluids (GNF), where we have systems for which the value of shear rate at a point within the fluid is determined only by the current value of shear stress at that point.
b) Viscoelastic fluids, where we have systems which exhibit a combination of viscous fluid behaviour and elastic solid-like behaviour, in which the relation between shear rate and shear stress is a time-dependent implicit relation.
c) Time-dependent fluids, where we have systems for which the relation between shear stress and shear rate shows further dependence on the duration of shearing and kinematic history.

This thesis focuses only on the first group of non-Newtonian fluids, the GNF, which can be classified into three categories: Bingham plastics, pseudoplastic fluids and dilatant fluids.

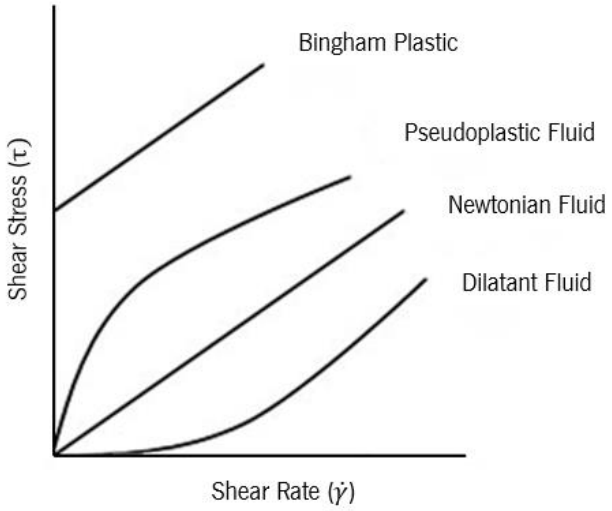

Figure 4 – Shear stress vs shear rate diagram for Newtonian and non-Newtonian fluids. Adapted from [14].

Bingham plastics are fluids that flow similar to a simple Newtonian fluid when shear stress exceeds the yield stress and behave like an elastic solid when shear stress is smaller than this value. Pseudoplastic fluids are characterized by a viscosity which generally falls continuously with an increase in shear rate, this is also known as the shear thinning effect. Very low and very high shear rate regions are the exceptions, where the flow curve is almost horizontal, as shown in Figure 5. This is currently the most commonly used non-Newtonian viscous fluid model to simulate the polymer flow. The viscosity of dilatant fluids increases with the increase of shear rate, this is also called the shear thickening effect. At low shear rates, shear thickening fluids also behave like a Newtonian fluid [7, 14].

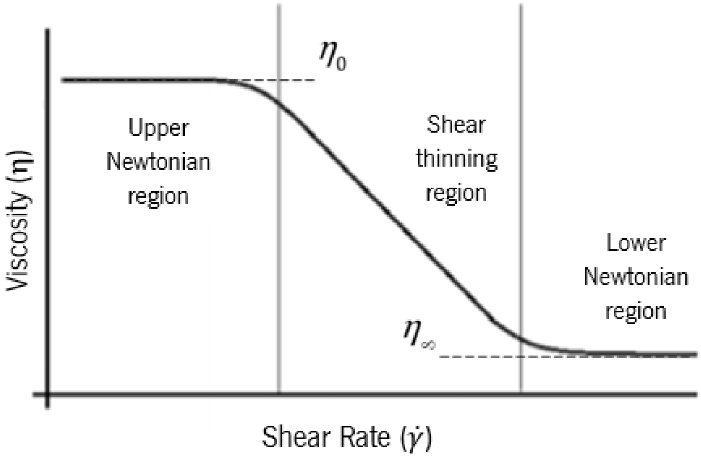

Figure 5 – Characteristic viscosity curve of pseudoplastic fluids. Adapted from [7].

The GNF model can describe the non-Newtonian characteristic that viscosity changes with the deformation rate tensor, but it cannot predict other phenomena such as recoil, elastic recovery, creep and extrudate swell, which are commonly observed in polymer processing flows [7].

All of these details can be considered as the result of the viscoelastic fluids exhibit both viscous and elastic characteristics [15]. Thus, the mathematical models of polymer processing should ideally be based on the use of viscoelastic constitutive equations [7]. However, the viscoelasticity of polymeric fluids is very complicated to be used on numerical simulations because it increases the simulation time. Thus, commercial codes can only work with GNF models and the same approach will be considered in this thesis.

The relationship between the viscous stress tensor $\boldsymbol{\tau}$ and the rate of strain tensor $\dot{\boldsymbol{\gamma}}$ is called a constitutive equation. A constitutive equation is a relation between two physical quantities and is specific to each material. The constitutive equations for Newtonian and generalized Newtonian fluids models will be presented hereafter.

2.2.1 Newtonian fluids

Fluids for which stress is linearly proportional to deformation, see Equation (3), are called Newtonian fluids.

$$\boldsymbol{\tau} = \mu\dot{\boldsymbol{\gamma}} \tag{3}$$

where $\boldsymbol{\tau}$ is the shear stress tensor, $\dot{\boldsymbol{\gamma}}$ is the strain rate or rate of deformation tensor and μ is the fluid dynamic viscosity. For such fluids, the viscosity remains constant, no matter the amount of shear applied for a constant temperature. Fluids for which Equation (3) does not hold are called non-Newtonian fluids [16]. Some examples of Newtonian fluids are water, mineral oil, gasoline and alcohol. Molten polymers, however, are non-Newtonian, and their viscosity, which is nonlinear in nature, can be very complicated to describe, depending on its chemical structure, composition, and processing conditions [7].

2.2.2 Generalized Newtonian fluids

As the molten polymer shows a variation of the viscosity with the deformation rate, Equation (3) needs to be changed in order to account for that behaviour. The simplest solution is to allow the dynamic viscosity μ to be a function of the rate of deformation [7]. Thus Equation (3) becomes

$$\boldsymbol{\tau} = \eta\left(\dot{\gamma}\right)\dot{\boldsymbol{\gamma}} \tag{4}$$

where $\eta\left(\dot{\gamma}\right)$ is the viscosity function, generally called viscosity. This equation is often referred to as the generalized Newtonian fluid model. Here the strain rate tensor $\dot{\boldsymbol{\gamma}}$ is defined by

$$\dot{\boldsymbol{\gamma}} = \nabla\mathbf{u} + \left(\nabla\mathbf{u}\right)^T = \begin{bmatrix} 2\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y}+\frac{\partial v}{\partial x} & \frac{\partial u}{\partial z}+\frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y}+\frac{\partial v}{\partial x} & 2\frac{\partial v}{\partial y} & \frac{\partial v}{\partial z}+\frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z}+\frac{\partial w}{\partial x} & \frac{\partial v}{\partial z}+\frac{\partial w}{\partial y} & 2\frac{\partial w}{\partial z} \end{bmatrix} \tag{5}$$

where $\mathbf{u}$ is the velocity vector and u,v and w are its components, $\mathbf{u} = (u, v, w)$. The effective shear rate $\dot{\gamma}$ is given by the second invariant of the strain rate tensor $\dot{\boldsymbol{\gamma}}$ which is

$$\dot{\gamma} = \sqrt{\frac{1}{2}\dot{\gamma}_{ij}\dot{\gamma}_{ij}} \tag{6}$$

where the summation convention is employed, i.e., whenever there are repeated subscripts, a summation is implied over the range of the subscripts, from (1, 2, 3).

Nowadays, the generalized Newtonian fluid model is widely used as the constitutive model to characterize the polymer used on the injection moulding simulation [7].

To model the injection moulding process, a viscosity function is required. The viscosity curves of most thermoplastic have the same dependence on shear rates as shown in Figure 6, in this figure is also shown the molecular alignment with viscosity variation. At upper Newtonian Region (lower shear rates), the viscosity is a nearly constant. At shear thinning region polymer chains are more uniformly aligned as the shear rates increase, so the viscosity decreases accordingly. When all polymeric chains are fully aligned, at the lower Newtonian region, viscosity becomes virtually insensitive to shear rate [7].

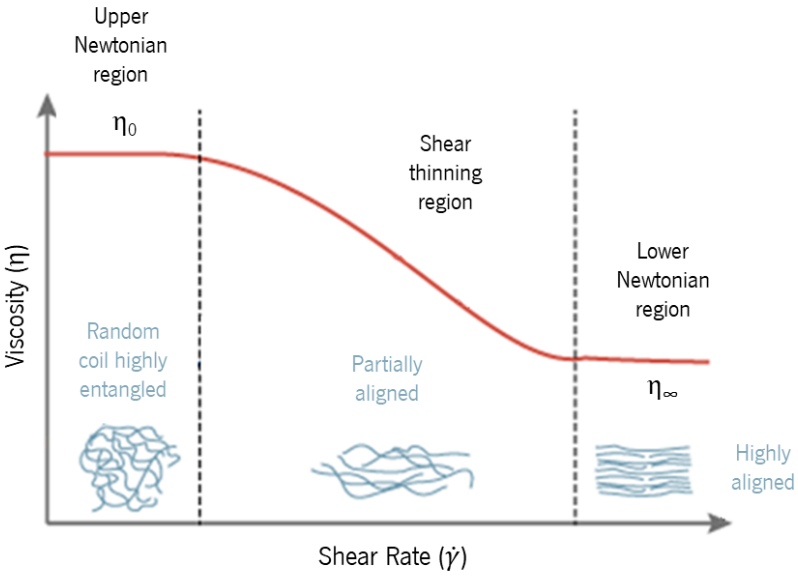

Figure 6 – Molecular alignment with viscosity variations, adapted from [7].

Several well-known models are available, so it is important to choose a model, which is both accurate over the processing range and for which data can be readily obtained. In the next points are presented the following viscosity models: power law model, Bird-Carreau model and Cross model.

- **Power law model**

The power law model can represent the behaviour of polymer melts at the high shear rate region, but it neglects the upper and lower Newtonian regions. The viscosity approximation by this model is shown in Figure 7.

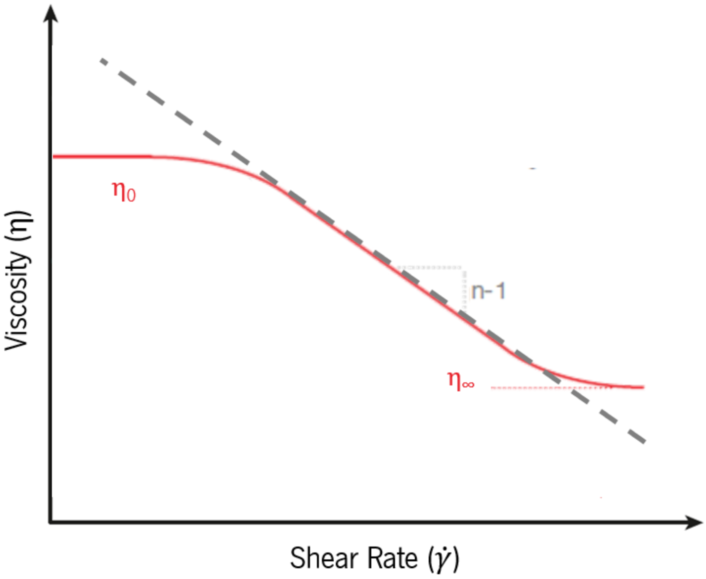

Figure 7 – Viscosity curve (red line) and approximation by the Power Law model (dashed line). Adapted from [17].

This model has the form:

$$\eta = \eta_0 \dot{\gamma}^{n-1} \tag{7}$$

where η_0 is the Newtonian viscosity or zero-shear rate viscosity; n is the power law index with a value between 0 and 1 (n = 1, recovers the Newtonian fluid behaviour).

- **Bird-Carreau model**

The Bird-Carreau model accounts for the observed Newtonian plateaus and fits a wide range of strain rates. The viscosity approximation using this model is shown in Figure 8.

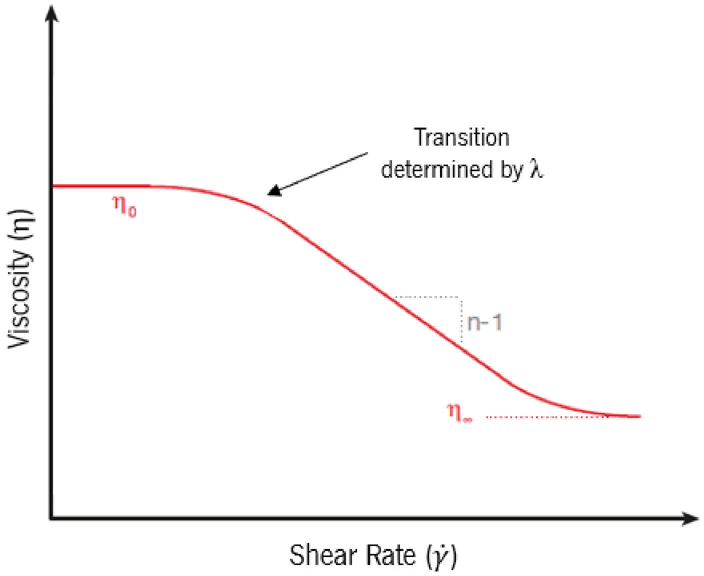

Figure 8 – Viscosity approximation using the Bird-Carreau model. Adapted from [17].

This model has four parameters and has the form

$$\frac{\eta - \eta_\infty}{\eta_0 - \eta_\infty} = \left[1 + (\lambda \dot{\gamma})^2\right]^{(n-1)/2} \tag{8}$$

where η_∞ is the viscosity at infinite shear rate; η_0 is the zero-shear rate viscosity, n is a dimensionless constant with the same interpretation as that in the previous equation and λ is a material time constant parameter.

- ### Cross model

The Cross model is similar to the Bird-Carreau model, this model describes both Newtonian and shear thinning behaviour.

The equation for the Cross model is given by

$$\eta = \frac{\eta_0}{1+\left(\eta_0 \dot{\gamma} / \tau^*\right)^{1-n}} \tag{9}$$

where τ^* is the shear stress at the transition between Newtonian and power law behaviour.

The zero-shear-rate viscosity, η_0, given in Equations (7), (8) and (9), contains the effect of temperature and pressure on viscosity, which can be described by several models depending on the specific material processed and the desired temperature range. In the field of plastic injection moulding process, the Arrhenius and Williams-Landel-Ferry (WLF) models are the most common [7].

- ### Arrhenius model

Arrhenius model, which applies to semicrystalline polymers, is given by

$$\eta_0 = B \exp\left(\frac{T_b}{T}\right) \exp\left(\beta p\right) \tag{10}$$

where B, T_b and β are material parameters. For amorphous thermoplastic the Arrhenius shift is valid for temperatures $T > T_g + 100\ ^{\circ}C$. Hence, for lower temperatures, the temperature dependence of the viscosity of amorphous thermoplastics is better described by the WLF model [17].

- ### WLF model

The WLF model has the form

$$\eta_0 = D_1 \exp\left(\frac{-A_1\left(T - T_c\right)}{A_2 + \left(T - T_c\right)}\right) \tag{11}$$

$$T_c = D_2 + D_3 p \tag{12}$$

in which A_1, A_2, D_1, D_2 and D_3 are material parameters.

When η_0 of Cross model is given by Equations (11) and (12) then Equation (9) is the so called Cross-WLF model. The viscosity approximation with this model is represented in Figure 9.

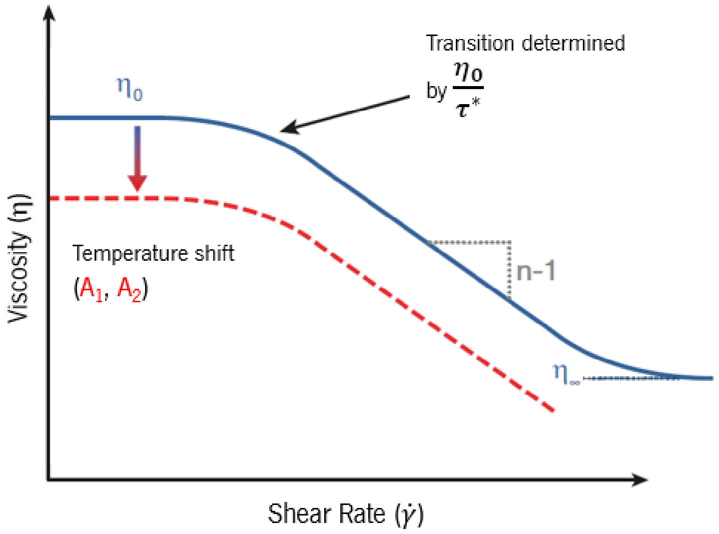

Figure 9 – Viscosity approximation with the Cross-WLF model. Adapted from [17].

The Cross-WLF model is the most common model used in injection moulding simulation software because it offers the best fit for most viscosity data [18].

2.3 THERMAL PROPERTIES

The thermal characteristics of polymers, such as melt temperature, glass transition temperature, thermal conductivity, specific heat and thermal diffusivity, influence their properties as well as their processability, and thus, a proper knowledge of the parameters that affect these properties is essential to establish the best process conditions to meet product performance requirements. The most important thermal properties for the injection moulding process are described hereafter.

2.3.1 Melt temperature

The melt temperature T_m is the lowest temperature from which it is possible to inject, and some common values are presented in Table 1. For semicrystalline polymers, it happens as a relatively sharp point and it occurs when there are no more crystalline regions. Amorphous polymers do not have a distinct T_m, they simply start melting as soon as the heat cycle begins [7].

Table 1 – Melt temperature values for some well-known semicrystalline polymers [10].

Polymer	Melt temperature, T_m (°C)
Polypropylene, PP	168
High-density polyethylene, HDPE	134
Polyamide, PA	260
Polyethylene terephthalate, PET	250

2.3.2 Glass transition temperature

The glass transition temperature T_g is also called the glass-rubber transition temperature, and in Table 2 are presented some common values of T_g for well-known polymers. Below this temperature, the polymer changes from a rubbery to a transparent hard glass-like behaviour. For temperatures greater than T_g, it is not as strong or rigid as glass, it is a viscous liquid. T_g is usually reported as a single value. However, it occurs over a temperature range and is kinetic in nature [7, 20].

Table 2 – Glass transition temperature values for some well-known semicrystalline and amorphous polymers [10].

	Polymer	Glass transition temperature, T_g (°C)
Semicrystalline	Polypropylene, PP	-10
	High-density polyethylene, HDPE	-110
	Polyamide, PA	50
	Polyethylene terephthalate, PET	70
Amorphous	Acrylonitrile butadiene styrene, ABS	102
	Polystyrene, PS	90
	Poly(methyl methacrylate), PMMA	100
	Polycarbonate, PC	150
	Polyvinyl chloride, PVC	90

2.3.3 Thermal conductivity

Thermal conductivity k is the property of a material that accounts for its ability to conduct heat. Low thermal conductivity values are required when the purpose is to minimize heat transfer. On the other hand, when heat transfer from one site to another is desirable, materials with higher thermal conductivity must be chosen [20]. Generally, polymers have a low thermal conductivity, and Table 3 shows some values for well-known polymers.

Thermal conductivity does not vary significantly among the thermoplastic materials, it is relatively insensitive to temperature and independent of molecular weight. Generally, the thermal conductivity of thermoplastics is relatively low when compared to metals [21].

When modelling the injection moulding process, the thermal conductivity can be considered constant or can be considered a linear interpolation function, these two options are shown below.

Table 3 – Thermal conductivity values for some well-known semicrystalline and amorphous polymers [22].

	Polymer	**Thermal conductivity k (W/(mK))**
Semicrystalline	Polypropylene, PP	0,12
	High-density polyethylene, HDPE	0,44
	Polyamide, PA	0,25
	Polyethylene terephthalate, PET	0,15
Amorphous	Acrylonitrile butadiene styrene, ABS	0,33
	Polystyrene, PS	0,14
	Poly(methyl methacrylate), PMMA	0,21
	Polycarbonate, PC	0,20
	Polyvinyl chloride, PVC	0,19

- ### Constant thermal conductivity model

The simplest model of the thermal conductivity is the constant thermal conductivity model, that is

$$k = k_0 \tag{13}$$

where k is the thermal conductivity and k_0 is its specified value. It assumes that the thermal conductivity is independent of temperature [21].

- ### Linear interpolation function of thermal conductivity

One common approximation of the dependence of thermal conductivity on temperature, comparable to specific heat capacity, is the linear interpolation function [7]. Given thermal conductivities, k_1 and k_2, at two different temperatures, T_1 and T_2, we can obtain the following linear equation

$$k = aT + b, a = \frac{k_2 - k_1}{T_2 - T_1}, b = \frac{T_2 k_1 - T_1 k_2}{T_2 - T_1} \tag{14}$$

2.3.4 Specific heat capacity

The specific heat capacity c_p, represents the amount of energy required to change the temperature of one unit mass of material by one degree [23]. Table 4 shows the values of the specific heat for some well-known polymers.

Similarly to thermal conductivity, in the case of specific heat capacity, when modelling the injection moulding process, the specific heat can be considered constant or following a linear interpolation function, these two options are shown below.

Table 4 – Specific heat capacity for some well-known semicrystalline and amorphous polymers *[10]*.

	Polymer	**Specific heat** c_p **(J/(kgK))**
Semicrystalline	Polypropylene, PP	3768
	High-density polyethylene, HDPE	3768
	Polyamide, PA	314
	Polyethylene terephthalate, PET	1884
Amorphous	Acrylonitrile butadiene styrene, ABS	2093
	Polystyrene, PS	2093
	Poly(methyl methacrylate), PMMA	2344
	Polycarbonate, PC	2093
	Polyvinyl chloride, PVC	2512

- **Constant specific heat capacity model**

Constant specific heat capacity model is in most of the commercial software the only available model, and it has the following form

$$c_p = c_{p0} \tag{15}$$

where c_p is the heat capacity and c_{p0} is a given value.

- **Linear interpolation function of specific heat capacity**

Another common approximation of the dependence of specific heat capacity on temperature is the linear interpolation function [7]. This approximation has the form:

$$c_p = aT + b, a = \frac{c_{p2} - c_{p1}}{T_2 - T_1}, b = \frac{T_2 c_{p1} - T_1 c_{p2}}{T_2 - T_1} \tag{16}$$

where c_{p1} and c_{p2} are given specific heat capacities, at two different temperatures, T_1 and T_2, respectively.

2.3.5 Thermal diffusivity

The thermal diffusivity D is the physical property of the material which governs the time-dependent temperature distribution. The thermal diffusivity plays an important role in determining the cycle time in the injection moulding of polymers [24].

The thermal diffusivity is related to the thermal conductivity and to the specific heat capacity through the following relation

$$D = \frac{k}{\rho c_p} \tag{17}$$

where ρ is the density of the polymer.

2.4 PRESSURE-VOLUME-TEMPERATURE BEHAVIOUR

One characteristic of polymers is their compressibility. When pressure is applied, they can compress significantly in proportion to the amount of pressure applied.

The Pressure-Volume-Temperature (PVT) behaviour or the thermodynamic material behaviour is the dependence of specific volume on pressure and temperature. This PVT behaviour has a significant influence on the process course in injection moulding, especially on the holding phase, and on the shrinkage and warpage of the final part. As the polymer is injected, it is not only hot and, therefore, expanded but also under significant pressure it reduces its volume. This makes it difficult to arrive at a true shrinkage factor because the actual change in volume depends on the type of polymer, the melt temperature, the injection pressure required to fill the cavity space, and the temperature at which it will be ejected from the mould [25].

The PVT behaviour of amorphous and semicrystalline polymers are different, Figure 10 shows the typical PVT diagrams of an a) amorphous and a b) semicrystalline polymer. PVT diagram describes the specific volume as a function of pressure and temperature. For both types of material, the specific volume in the melting range changes linearly with temperature. As pressure increases, the specific volume decreases. In the temperature range where they are solids, these two types of polymers differ: amorphous plastic have a linear dependence of specific volume on temperature, but the semicrystalline counterparts have an exponential dependence [7]. When crystalline materials are cooled below their transition temperature, the molecules arrange themselves in a more orderly way, forming crystallites. The process of crystallization results in a more orderly way and hence denser packing for semicrystalline materials, and the specific volume in the solid range depends on the cooling rate [26].

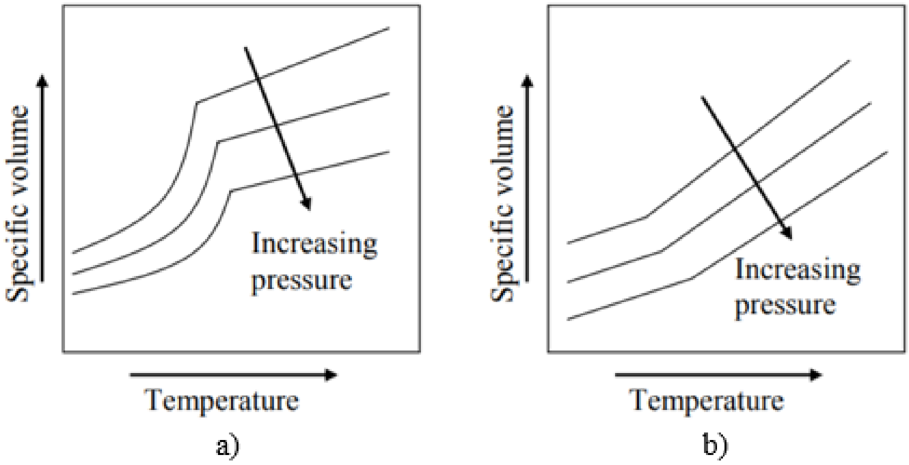

Figure 10 – Illustration of PVT-diagram for semicrystalline a) and amorphous polymers b). The specific volume is plotted as a function of the temperature and for different pressures. Adapted from [27].

Semicrystalline polymers have one significant and abrupt change of volume. However, in case of amorphous polymers, only a slight change happens when shifting from the molten state into a solid one.

A good PVT model should, therefore, be able to characterise this difference between amorphous and semicrystalline polymers and still the specific volume dependency with temperature and pressure [28].

An equation which describes this behaviour is called equation of state, and some of the most relevant ones will be presented hereafter.

- **Constant specific volume**

When dealing with incompressible materials, the model of constant specific volume should be used since there is no variation of the specific volume with pressure and temperature [29], i.e.,

$$\rho = \rho_0 \tag{18}$$

where ρ_0 is the material density.

- **Spencer-Gilmore model**

This model is derived from the law of ideal gases and adds one correction term of pressure and temperature to the specific volume term [7]. The model has the following form:

$$\hat{V} = \hat{V}_0 + \frac{\hat{R}T}{p + p_0} \tag{19}$$

where $\hat{V}$ is the material specific volume, $\hat{V}_0$ is the reference value of material specific volume on a specific condition and T is the temperature of the material [29]. There are three parameters to be determined to employ this model: $\hat{V}_0$, $\hat{R}$ and p_0.

○ Tait equation

The Tait equation is purely empirical and was originally proposed for water [30]. This equation has the following form:

$$\hat{V} = \hat{V}_0 \left[1 - C \ln \left(1 + \frac{p}{B} \right) \right] \tag{20}$$

$$\hat{V}_0 = b_1 \exp\left(-b_2 T \right) \tag{21}$$

$$B = b_3 \exp\left(-b_4 T \right) \tag{22}$$

with five parameters to be determined b_1, b_2, b_3, b_4 and C, to characterize the material. This model does not consider the abrupt change of volume of the semicrystalline polymers [29].

○ Modified Tait equation

As the original Tait model cannot handle the abrupt change of volume of the semicrystalline polymers, thus a modification was proposed to be able to describe the PVT relation of both semicrystalline and amorphous polymers [7]. The modified Tait equation has the following form:

$$\hat{V} = \hat{V}_0 \left[1 - C \ln \left(1 + \frac{P}{B} \right) \right] + \hat{V}_t \tag{23}$$

$$\hat{V}_0 = \begin{cases} b_{1S} + b_{2S} \bar{T} \ \textbf{if} \ T \le T_t \\ b_{1L} + b_{2L} \bar{T} \ \textbf{if} \ T > T_t \end{cases} \tag{24}$$

$$B = \begin{cases} b_{3S} \exp(-b_{4S} \bar{T}) \ \textbf{if} \ T \le T_t \\ b_{3L} \exp(-b_{4L} \bar{T}) \ \textbf{if} \ T > T_t \end{cases} \tag{25}$$

$$\hat{V}_t = \begin{cases} b_7 \exp(b_8 \bar{T} - b_9 p) \ \textbf{if} \ T \le T_t \\ 0 \qquad\qquad\qquad \textbf{if} \ T > T_t \end{cases} \tag{26}$$

$$\bar{T} = T - b_5 \tag{27}$$

$$T_t = b_5 + b_6 p \tag{28}$$

where $C = 0{,}0894$ and it requires the definition of 13 parameters to characterize a specific material. In the above relation, Equation (26) characterizes the semicrystalline polymers abrupt change of volume when these are at an approximate temperature of its molten state [29]. T_t is used to characterize the abrupt change of viscosity near the transition temperature [29]. This equation of state is the most convenient and the most widely used for polymers [26].

3. GOVERNING EQUATIONS FOR THE INJECTION MOULDING PROCESS

The governing equations of the fluid flow, namely the equations of mass conservation, momentum and energy, employed to simulate the injection moulding process, will be presented in this section. The derivation of these equations will not be presented, but can be found in several books [4, 32].

3.1 MASS CONSERVATION EQUATION

The differential equation expressing the conservation of mass is

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \mathbf{u} \right) = 0 \tag{29}$$

where ρ is the density, t is the time and $\mathbf{u}$ is the velocity vector. This equation is also known as the continuity equation. The first term on the left-hand side (LHS) is the rate of change of the mass per unit volume, the density. The second term describes the net flow of mass out of the element across its boundaries and is called the convective term. For an incompressible fluid, the density is constant and so the continuity equation reduces to

$$\nabla \cdot \mathbf{u} = 0 \tag{30}$$

3.2 MOMENTUM CONSERVATION EQUATION

Newton’s second law states that the rate of change of momentum of a fluid particle equals the sum of the forces on the particle. The vectorial equation of the conservation of momentum is

$$\frac{\partial \left(\rho \mathbf{u} \right)}{\partial t} + \nabla \cdot \left(\rho \mathbf{u} \mathbf{u} \right) = \nabla \cdot \left(\boldsymbol{\sigma} \right) + \rho \mathbf{f} \tag{31}$$

where $\boldsymbol{\sigma}$ is the total stress tensor and $\mathbf{f}$ represents body forces.

The LHS of the equation of the conservation of momentum equation represents the rate of change of momentum of a fluid particle, which is given by the sum of the rate of increase of momentum of fluid a element (first term of the LHS) and the net flow rate of momentum out of a fluid element (second term of the LHS). The first and second terms of the right-hand side (RHS) account respectively, for the total force on the element due to surface and body forces.

The total force on the element due to surface stresses is

$$\nabla \cdot \left(\boldsymbol{\sigma} \right) = -\nabla p + \nabla \cdot \boldsymbol{\tau} \tag{32}$$

where the first term on the RHS is related to pressure and the second term to the deviatoric stresses.

For the incompressible case, the conservation of momentum equation becomes

$$\frac{\partial(\rho\mathbf{u})}{\partial t}+\rho\mathbf{u}\cdot\nabla\cdot\mathbf{u}=\nabla\cdot(\boldsymbol{\sigma})+\rho\mathbf{f} \tag{33}$$

Dividing all the terms by ρ, the conservation of momentum equation for the incompressible case becomes

$$\frac{\partial(\mathbf{u})}{\partial t}+\mathbf{u}\cdot\nabla\cdot\mathbf{u}=-\nabla P+\frac{\nabla\cdot\boldsymbol{\tau}}{\rho}+\mathbf{f} \tag{34}$$

where P is the modified pressure $P=\frac{p}{\rho}$.

3.3 ENERGY CONSERVATION EQUATION

The energy conservation equation is based on the first law of thermodynamics, which states that the rate of energy of a fluid particle is equal to the rate of heat addition to the fluid particle plus the rate of work done on the particle.

Internal energy conservation equation is given by

$$\frac{\partial(\rho i)}{\partial t}+\nabla\cdot(\rho i\mathbf{u})=-p\cdot(\nabla\cdot\mathbf{u})+\nabla\cdot(k\cdot\nabla T)+S_i+$$
$$\left[\tau_{xx}\frac{\partial u}{\partial x}+\tau_{yx}\frac{\partial u}{\partial y}+\tau_{zx}\frac{\partial u}{\partial z}+\tau_{xy}\frac{\partial v}{\partial x}+\tau_{yy}\frac{\partial v}{\partial y}+\tau_{zy}\frac{\partial v}{\partial z}+\tau_{xz}\frac{\partial w}{\partial x}+\tau_{yz}\frac{\partial w}{\partial y}+\tau_{zz}\frac{\partial u}{\partial z}\right] \tag{35}$$

where i is the internal energy, k is thermal conductivity, τ_{ij} are the viscous deviatoric stress components and S_i is the source term.

The LHS represents the rate of change of internal energy of a fluid particle, the first term is the rate of increase of internal energy of a fluid element and the second term is the net rate of flow of internal energy out of fluid element. The first term and the terms inside brackets on RHS represent the total rate of work done on particle by surface forces. The second term on the RHS is the rate of heat addition to the fluid particle due to heat conduction.

For the incompressible case ($\nabla \cdot \mathbf{u} = 0$), the density ρ is constant, and considering the internal energy $i = c_p T$ in which c_p is the specific heat capacity, the energy conservation equation, Equation (35) becomes

$$\begin{gathered}\rho c_p \frac{\partial (T)}{\partial t} + c_p \nabla \cdot \left(\rho T \mathbf{u} \right) = \nabla \cdot \left(k \cdot \nabla T \right) + S_i + \\ \left[\tau_{xx} \frac{\partial u}{\partial x} + \tau_{yx} \frac{\partial u}{\partial y} + \tau_{zx} \frac{\partial u}{\partial z} + \tau_{xy} \frac{\partial v}{\partial x} + \tau_{yy} \frac{\partial v}{\partial y} + \tau_{zy} \frac{\partial v}{\partial z} + \tau_{xz} \frac{\partial w}{\partial x} + \tau_{yz} \frac{\partial w}{\partial y} + \tau_{zz} \frac{\partial u}{\partial z} \right]\end{gathered} \tag{36}$$

This equation can be simplified dividing all the terms by ρc_p, Equation (36) becomes

$$\begin{gathered}\frac{\partial (T)}{\partial t} + \nabla \cdot \left(T \mathbf{u} \right) = \Omega \nabla^2 T + \frac{S_i}{\rho c_p} \\ \frac{1}{\rho c_p} \left[\tau_{xx} \frac{\partial u}{\partial x} + \tau_{yx} \frac{\partial u}{\partial y} + \tau_{zx} \frac{\partial u}{\partial z} + \tau_{xy} \frac{\partial v}{\partial x} + \tau_{yy} \frac{\partial v}{\partial y} + \tau_{zy} \frac{\partial v}{\partial z} + \tau_{xz} \frac{\partial w}{\partial x} + \tau_{yz} \frac{\partial w}{\partial y} + \tau_{zz} \frac{\partial u}{\partial z} \right]\end{gathered} \tag{37}$$

where $\Omega = \dfrac{k}{\rho c_p}$.

4. NUMERICAL METHODS

To use mathematical models in a computer, a numerical method is required. Only in the most simplest cases it is possible to find exact analytical solutions for the governing equations in the model, thus, in general, one has to rely on numerical techniques for finding approximate solutions [32].

There are several numerical methods available to solve the governing equations which characterize the three-dimensional flow in the injection moulding process machine, such as the finite-difference method (FDM), the finite-element method (FEM), the boundary element method (BEM) and the finite-volume method (FVM). These numerical methods generally require discretization of the entire physical and temporal domains into finite sub-domains and small time steps, respectively. The governing equations are discretized into a system of algebraic equations to be solved by use of the computer [33]. The FEM and FVM are the prevailing numerical methods for three-dimensional simulation of injection moulding due to their capability to deal with complex geometries and non-linear problems. Below each one of these numerical methods will be briefly presented.

4.1 FINITE-DIFFERENCE METHOD

The FDM is a relatively efficient and simple numerical method for solving partial differential equations. In this method, the physical domains are discretized in the form of finite-difference grids. At each grid point, the differential equations are approximated by replacing the partial derivatives by approximations in terms of the nodal values of the function, and a set of algebraic equations is then generated [34], [35].

The resulting algebraic equations are then solved numerically.

The FDM is very simple and effective for simple geometries, the disadvantage of the FDM is that the conservation equation is not enforced unless special care is taken. Also, the restriction to simple geometries is a significant disadvantage in complex flows [35, 36]. It is difficult apply FDM to injection moulding since it is typical to have a highly irregular boundary or complex domains [34], [35].

4.2 FINITE-ELEMENT METHOD

FEM requires discretizing the physical domain into finite elements. The field variables are represented with shape functions and nodal values over each finite element. Using residual minimization techniques, the governing equations are transformed into discretized forms [35, 37].

The FEM is convenient for complex mathematical models in complicated geometrical models, and, therefore FEM can handle irregular geometries with different boundary conditions and material behaviours, which is the main advantage of this method [37]. One disadvantage is that for three-dimensional simulation of injection moulding it requires large memory and CPU time [35, 37].

4.3 BOUNDARY-ELEMENT METHOD

In the BEM only the boundary of the domain needs to be discretized. The discretization is derived through application of the weighted-residual method, with the weighting function taken to be the fundamental solution, the integration by parts converts the volume integral of the residual to surface integrals and the single point function values are used. Consequently, the number of unknowns and the effort for computation and mesh generation are reduced significantly, which is the main advantage of this method. The resulting matrix systems of algebraic equations are dense and non-symmetric. However, when the fundamental solution cannot be easily derived as in the cases of non-linear problems, the utility of this method is severely impaired. In addition, the formulation is significantly more involved than that of FEM and FDM [39–41].

4.4 FINITE-VOLUME METHOD

The finite volume method (FVM) is a numerical technique that transforms the partial differential equations representing conservation laws over differential volumes into discrete algebraic equations over finite volumes. The first step in the solution process is the discretization of the geometric domain, which, in the FVM, is divided into small control volumes. The partial differential equations are then transformed into algebraic equations by integrating them over each control volume. The system of algebraic equations is then solved to compute the values of the dependent variable for each of the elements [41].

In the FVM, some of the terms of the conservation equation are transformed into face fluxes and evaluated at the control volume faces. Since the integral forms of equations are solved directly in the physical domain mass, momentum and energy are automatically conserved. This inherent conservation property of the FVM makes it the preferred method in computational fluids dynamics (CFD). Another important attribute of the FVM is that it can be formulated in the physical space on unstructured polygonal meshes. Finally, in the FVM it is quite easy to implement a variety of boundary conditions in a non-invasive manner since the unknown variables are evaluated at the centroids of the volume elements, not at their boundary faces [41].

These characteristics have made the FVM quite suitable for the numerical simulation of a variety of applications involving fluid flow and heat and mass transfer, and developments in the method have been closely linked with advances in CFD [41].

CHAPTER III – STATE OF THE ART

1. INTRODUCTION

The injection moulding process is one of the most important technique for the production of polymeric parts, in which competition has been pushing the industry to increase parts quality at the lowest possible cost. However, the quality and performance of injection-moulded parts depend on the material, part and mould design and process variables. This complex framework motivated the development of reliable modelling codes, which started on the second half of the last century with the first attempts devoted to simulating the filling of a cavity in order to obtain the fluid velocity profiles [42]. The first attempts to model the injection moulding were conducted by Spencer and Gilmore [43–46], in 1950, in which they considered a simple one-dimensional mathematical model.

The demand for increased quality modelled parts in the 1970s, reinforced the interest in developing mathematical models capable of simulating the injection moulding process. At that time, only the filling stage of the injection moulding process was considered. Since 1978, commercial Computer Aided Engineering (CAE) software has been available, over the years, the scope of such software has expanded beyond filling analysis to include cooling analysis, part gate location, runner sizing, weld line prediction, gas-trap prediction, warpage and residual stress analysis [47].

This state of the art chapter is divided into two parts, one which is devoted to the complexity of the mathematical models used to describe the injection moulding process, going from the one-dimensional (1D) to the three-dimensional (3D) models and the second part concerning the available software created for injection moulding, covering both proprietary and open-source software.

2. MATHEMATICAL MODELS USED TO DESCRIBE THE INJECTION MOULDING PROCESS

2.1 ONE DIMENSIONAL APPROACHES

As discussed in the introduction, the first modelling studies devoted to the injection moulding process were presented in 1950 by Spencer and Gilmore [43–46] and were concerned with flow visualization of mould filling, modelling of the fluid mechanics of the process, pressure-temperature variations during the moulding cycle, and orientation and residual stresses in moulded parts. In their mould filling studies, they considered a simple 1D model and fitted a power law type equation relating filling time to the applied pressure drop.

A second generation of studies were presented by Ballman and colleagues [48–51] with the first non-isothermal model for the filling of rectangular cavities with a small thickness and the first computer application for predictive calculations. Toor and Ballman *et al.* [51] introduced a scheme to calculate the average velocity of a polymer melt filling a cold rectangular cavity and obtained the maximum flow length of the polymer. These results could then be used to deduce the time to fill a cavity of a given length. Their calculations accounted for conductional heat loss and used experimentally determined parameters for the effect of temperature and shear rate on viscosity. No viscous dissipation effects were accounted for, and the pressure equations solved were obtained by a force balance. In their study, the equations were solved on an IBM 73 computer with an average run time per simulation of 20 hours. Later, Harry and Parrot [52], Lord and Williams [53] and Thienel and Menges [54] also studied the 1D filling behaviour in rectangular cavity geometries.

Harry and Parrot [52], in 1970, predicted the fill lengths and fill times of thin constant cross-section cavities, with a simulation model that considers heat conduction and viscous heat dissipation, along with the temperature dependence of the material properties. The sensitivity of the simulation tools was assessed by performing experimental trials with two different cavity thicknesses. The thinner cavity illustrated a short shot in all cases, while the thick cavity allowed or complete filling. The simulation tool accurately distinguished between the short shot and fill conditions, although significant error was identified for the length prediction on the short shot and the time-to-fill of the full shot condition.

Lord and Williams [53] described a practically-oriented computer model which computes the temperature, pressure and velocity fields in a cavity during the mould filling stage of the injection moulding process. The model can be used for cavities having a complex geometry and for commonly used materials with complex viscosity, shear rate and temperature relationships. Predictions of the model have been tested against exact solutions to special cases and found to be in good agreement. In moulding problems, model predictions have been found to correctly describe trends such as an increase in the pressure required to fill moulds as injection rate, shot temperature, and mould temperature decrease, and to be reasonably accurate when compared to data for plaque, disc, and telephone housing moulds over a wide range of moulding conditions.

Williams and Lord [55], and Nunn and Fenner [56] worked with 1D circular flow of polymer melts.

Williams and Lord [55] developed a finite difference analysis which predicts, the temperature, pressure and velocity distributions for the flow of thermoplastic materials in straight and tapered, hot and cold circular flow channels. The analysis when combined with the cavity analysis described in [53], allow the injection moulding process to be modelled from the shot to the cavity during injection. The analysis obtained good agreement for temperature and pressure distributions between model predictions and analytical solutions to special cases. The agreement between model predictions and experiment for surface temperature and pressure distribution were found to be satisfactory.

Nunn and Fenner [56] developed a method for the analysis of transient heat transfer and flow of viscous polymer melts in the nozzles of injection moulding machines. Melt viscosities were treated as non-Newtonian and sensitive to changes in both temperature and pressure, the constitutive equation used for viscous polymer melts was the power law. Typical results had shown that the flows become steady in a few milliseconds. On the other hand, they were very far from being thermally fully developed when flow leaves injection nozzles and entering moulds.

Kamal and Kenig [57, 58], Wu *et al.* [59], Stevenson [60] and Stevenson and Chuck [61] developed mathematical models to describe the filling in a centre-gated disc.

Kamal and Kenig [57] proposed a theoretical model which describes the behaviour of thermoplastics inside an injection moulded cavity. The model took into consideration the non-Newtonian behaviour of the melt, the effect of temperature on density and viscosity, the latent heat of solidification and the differences in thermal properties between the solid and the melt. Numerical solutions have been obtained for the case of spreading radial flow in a semi-circular cavity. The numerical results yield significant data on the progression of the melt front, the flow rate and the velocity profiles at different times and positions in the cavity. They also yield temperature and pressure profiles throughout the packing and cooling stages. To check the validity of the proposed model, the same authors [58] performed experimental studies, and the data collected show good agreement with theory for the injection moulding of polyethylene in a semi-circular cavity. The main problems in the application of the model occurred during the filling phase near the cavity entrance where viscoelastic and end effects were important and where some of the assumptions inherent in the model may not be valid.

Wu *et al.* [59] used the transport equations for a power-law fluid to solve the transient and non-isothermal problem of filling a disk-shaped cavity. Using the results obtained it was possible to predict gate pressures, fill times and short shots. Furthermore, the velocity and temperature fields were available throughout the filling process, and showed the presence of a frozen surface layer near the mould wall. Rigid PVC was primarily used in the simulations, but some results were also given for linear PE, PA66, ABS and PS.

Stevenson [60] developed a graphical method for estimating the injection moulding pressure and clamping force for the filling of disk-shaped mould cavities of constant thickness with amorphous polymers. The results reported were based on a numerical simulation of a power law fluid filling a cold mould at a constant injection rate. The design charts were based on a comparison of the numerically predicted non-isothermal injection pressure to an analytically calculated isothermal reference injection pressure and for the dimensionless bulk temperature as functions of the power-law exponent, Brickman number, dimensionless fill time and a ratio of temperatures. Relatively few injection moulded parts are filled through single gates or have surfaces in the shape of a circle, a geometric approximation is necessary order to apply this analysis to more typical mould geometries. Therefore to extend this work, Stevenson and Chuck [61] introduced geometric and semicrystalline material approximations. In comparisons with experimental data and more exact numerical predictions, these approximations and the radial flow simulation, known collectively as the Radial Flow Method (RFM), were shown to give satisfactory agreement for cavities filled through centre, off-centre and multiple gates with semicrystalline and amorphous materials. The results obtained with the RFM were in good agreement with those obtained by finite element simulations of two-directional flow.

These filling models are all limited to one-dimensional flow approximations. Considering that typically the cavities of the moulds are complex, to apply these one-dimensional flow representations, the branching flow [62] approach was proposed and implemented. This approach involves lying flat the geometry and decomposing it as a series of simple geometries such as strips, discs, fans, and/or tubes. The solution accuracy strongly depends on how the geometry is being branched which is unknown *a priori* and requires judgment from the user. To avoid the user judgement about the mould geometry, the following subsection presents two-dimensional approaches able to describing the behaviour of polymers inside complex injection moulded parts.

2.2 TWO-DIMENSIONAL APPROACHES

To overcome the deficiency of the, above mentioned, branching flow approach theoretical studies for 2D flow in a thin cavity based on the Hele-Shaw flow formulation have been conducted by Broyer and Tadmor *et al.* [63, 64], Krueger and Tadmor [65], Kamal *et al.* [66, 67], Hieber and Shen [68] and Shen [69].

The Hele-Shaw model is used for shear flow analysis in a thin cavity. The geometry of the solid part is simplified into a mid-plane model or the surface model [7].

This model considers an incompressible, generalized non-Newtonian fluid under non-isothermal conditions [70].

In this flow, as shown in Figure 11, the gap thickness between two plates is assumed to be much smaller than the other dimensions [71].

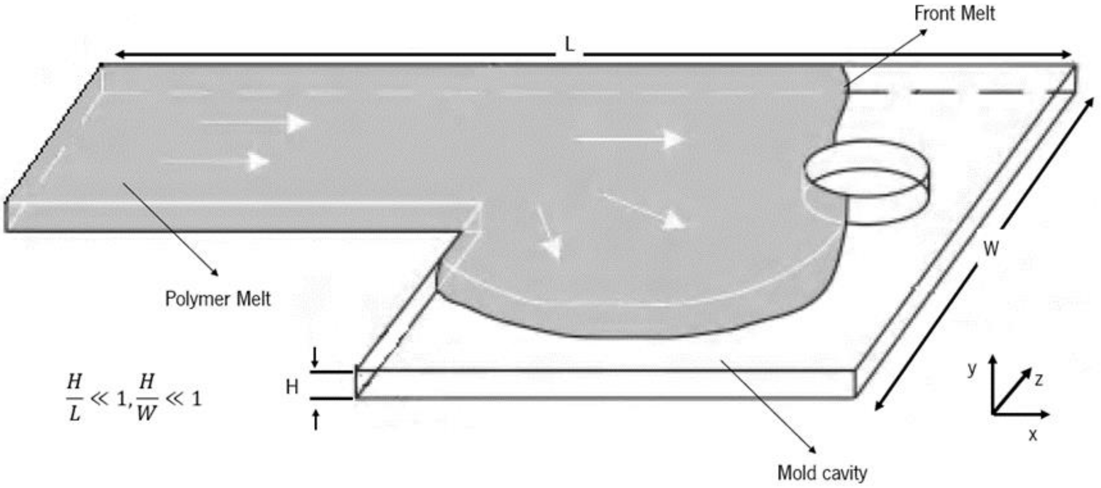

Figure 11 – Schematic diagram of a typical flow described by the Hele-Shaw model. Adapted from [4].

In this assumption, the pressure is a function of a planar coordinates only and the velocity in the gapwise direction is ignored. As a result, the continuity and momentum equations for polymer melt flow in injection moulding can be simplified into a single Poisson-like equation in terms of pressure and fluidity [68].

Accordingly with [72], the relevant governing equations describing the Hele-Shaw polymer melt flow are:

$$\frac{\partial \rho}{\partial t}+\frac{\partial}{\partial x}(\rho u)+\frac{\partial}{\partial y}(\rho v)+\frac{\partial}{\partial z}(\rho w)=0 \quad (38)$$

$$\frac{\partial p}{\partial x}=\frac{\partial}{\partial z}\left(\eta\frac{\partial u}{\partial z}\right) \quad (39)$$

$$\frac{\partial p}{\partial y}=\frac{\partial}{\partial z}\left(\eta\frac{\partial v}{\partial z}\right) \quad (40)$$

$$\rho c_p\left(\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}\right)=k\frac{\partial^2 T}{\partial z^2}+\eta\dot{\gamma}^2 \quad (41)$$

where x and y denote the planar coordinates and z the thickness coordinate, (u,v,w) are the velocity components in the (x,y,z)directions for time t and under pressure p and the shear rate $\dot{\gamma}$ is given by

$$\dot{\gamma}=\sqrt{\left(\frac{\partial u}{\partial z}\right)^2+\left(\frac{\partial v}{\partial z}\right)^2} \quad (42)$$

Broyer, Tadmor and Gutfinger [63] suggested the Flow Analysis Network (FAN) method, which is a simplified version of the general FEM. This method reduced the problem solving a set of linear algebraic equations, and it imposes no restrictions as to the complexity of the boundaries, including flow around obstructions such as spider legs as well as arbitrary die geometries. The method can be used to predict flow under different operating conditions or flow paths in injection moulding dies, moreover by trial-and-error simulation mould geometries were modified to reach a certain prescribed goal. The predicted results for agreed well with experimental data obtained for the filling of a rectangular cavity with Newtonian fluid.

Broyer, Gutfinger and Tadmor [64], adapted the previously developed flow analysis network (FAN) method [63] to the problem of the cavity filling process in injection moulding. The method is applicable to relatively narrow gap cavities of any shape, and it allows the computation of the advancing melt front at any instant of the process, as well as the prediction of the weld-line location. The method considered non-isothermal flow analysis which allowed prediction of the possibility of a short-shot situation.

Kamal *et al.* [66] extended their numerical model for the injection moulding process of simple semi-circular cavity [57, 58], to analyse and study the behaviour of commercial polymers in rectangular cavities. The extension to rectangular cavities was referred as Bounded Radial Flow Model (BRFM) and it is successful in describing the behaviour of the polymer, the model successful differentiates between the filling behaviour of various commercial polyethylene resins. Furthermore, the application of the model to the overall moulding process yields pressure profiles which were in very good agreement with experimental data. They verified that the model also predicts the substantial differences in moulding behaviour of the two resin having different density and flow characteristics. Kamal *et al.* [66] found that although the BRFM is successful in predicting the overall moulding behaviour of plastics, it suffers from two deficiencies. Firstly, the shape of the melt front is assumed based on experimental data and the second deficiency relates to the assumptions involved in analysing the packing stage where only average pressures may be estimated.

Kuo and Kamal [67] developed an analytical-numerical method, using a generalized viscous model, for determining the shape of a progressing flow front in a rectangular cavity and for computing flow quantities and temperature distributions. They showed that the use of a generalized viscous model can closely simulate the behaviour of thermoplastics in injection filling thin cavities. Besides, the pressure and the stream function can be described by the Laplace equations in the case of both viscosity and temperatures varying strongly across a narrow cavity but weakly in the flow direction. Good agreement was obtained between computational results and experimental measurements on pressure distribution and progression of flow front.

To simulate polymer flow in two-dimensional conditions two approaches have been proposed, namely, the network flow [65] and the two-step “predictor-corrector” [68] method. The network flow discretizes the cavity geometry into a network of rectangular elements, being the melt-front pattern, which defines the flow domain, calculated based on the velocity at the advancing front [72].

The two-step “predictor-corrector” approach is similar to the previous approach except that it constantly creates new finite elements at the advancing flow front by taking into account the melt-front velocity and the actual geometry. Although the two-step “predictor-corrector” could handle more complex geometries, it had a major drawback related to the required intensive user intervention [72].

Krueger and Tadmor [65] measured the advancing melt front profiles, pressures and temperatures during the filling of a rectangular cavity with several shaped inserts. The purpose was to elucidate the overall flow pattern in complex, variable gap rectangular cavities containing various shaped inserts. The experimental data collected were compared with numerical results obtained by the FAN computation method developed by Broyer, Gutfinger and Tadmor [64]. Krueger and Tadmor were encouraged to investigate, with the theoretical model, the effect of rheological properties on advancing front profile since they obtained a good agreement between the numerical front profiles and the experimental ones. Those experiments were used injection moulding conditions, in such a way that the isothermal flow assumption was valid. They found that the use of Newtonian fluids provides very good estimates on overall flow patterns and, thus, also on weld line locations and orientation distribution.

Subsequently, Hieber and Shen [68] presented a detailed formulation for simulating the injection moulding filling of thin cavities of arbitrary planar geometry. A hybrid numerical scheme was employed in which the planar coordinates were described in terms of finite elements and the gap-wise and time derivatives were expressed in terms of finite differences. The modelling was in terms of generalized Hele-Shaw flow for an inelastic, non-Newtonian fluid under non-isothermal conditions. The simulation was applied to the filling of a two-gated plate mould having an intentionally unbalanced runner system and the predictions were found to compare favourably with experimental data in terms of advancing melt-front patterns, weld line formation and pressure evolution at several prescribed points in the cavity.

Hieber and Shen [68, 69, 71] and Wang *et al* [74] employed a finite-element/finite difference scheme for simulating the filling of thin cavities with general planar geometry. This method discretizes the sprue/runner/cavity geometry into a network of 1D tubular elements and 2D triangular thin-shell elements. The user intervention was eliminated because the advancement of the melt front is automatically tracked by computing the filled volume fraction of each control volume associated with the nodes [72]. Several commercial software packages and research code used this approach as the standard numerical framework.

Shen [69] presented an outline for a FEM algorithm able to tackles of two key problems found on the simulation of the injection moulding process: the first one was the cavity filling problem and the second was the steady viscoelastic flow in the juncture region where sudden changes of the geometry and large strain rates occurs. To solve the sample juncture problem, the FEM program incorporating the Leonov viscoelastic model was able to solve flow with a Deborah number exceeding 100. The developed code was a major breakthrough for realistic predictions of features as the juncture pressure loss, maximum stress and orientation, which are important to industrial applications.

The Hele-Shaw flow approach was extended or incorporated by other researchers to simulate the mould cooling [75], residual stresses [76], fibre orientation [77], and shrinkage and warpage [78], [79], but also in different types of moulding processes as co-injection moulding [80], gas-assisted injection moulding [81], microchip encapsulation [82], injection/compression moulding [83], reaction injection moulding and the resin transfer moulding [84].

The basis of the hybrid finite element/finite difference numerical solution of the generalized Hele-Shaw flow, for the compressible viscous fluid under non-isothermal conditions, was developed by Chiang, Hieber and Wang [85], where a unified simulation model for the filling and post-filling stages was proposed.

For verification purposes, Chiang, Hieber and Wang [86] employed in their study two mould geometries of uniform cavity thickness, and the material employed include both amorphous and semicrystalline polymers. For the two test moulds used and several processing conditions employed, the predicted pressure history over the entire filling and post-filling stages were found to be in good agreement with the corresponding experimental data. The assumptions of constant density and thermal properties, as well as an Arrhenius-type temperature sensitivity of the shear viscosity, were found to be adequate for the filling stage, but for the post-filling stage more accurate representations for all these properties were required. In order to compensate for the shrinkage due to continuous cooling, additional material was packed into the cavity, so the polymer compressibility was a critical ingredient in modelling the material behaviour during the post-filling stage. Moreover, it was crucial to incorporate an accurate temperature dependence of the thermal properties and shear viscosity as a result of large temperature variations.

Zhou *et al* [85–87] presented the first integrated mathematical model of the surface model, where it represents a three-dimensional part with a boundary mesh instead of the mid-plane. However, but it still adopts the Hele-Shaw assumption and therefore its formulation was very similar to that of the mid-plane model. The predicted results were found to be in good agreement with the corresponding experimental data and C-Mold results. Nowadays, this approach is still used by a large number of mould manufacturers and injection moulding polymers converters, so it was a huge step forward for the development of the injection moulding technology.

2.3 THREE-DIMENSIONAL FORMULATION

Hele-Shaw flow formulation has its limitations owing to the inherent creeping-flow, and thin-wall assumptions. The Hele-Shaw cannot accurately simulate the three-dimensional (3D) flow behaviours within thick and complex components or at the free surface, near and at the solid walls, and at the merging of two or more fluid streams [33, 88, 89]. Additionally, it cannot correctly predict the three-dimensional flow behaviours at flow junctions, regions where the thickness sudden changes or separate melt fronts meet together, and regions around special part features such as ribs, bosses or corners [7]. Hence, to generate complementary and more detailed information related to the flow characteristic and stress distributions in thick-moulded parts, a 3D simulation modelling approaches were developed.

Up to date, several 3D simulation approaches for the injection moulding process have been proposed. The first to appear was based in the FEM, Rajupalem *et al* [92] and Kim and Turng [93] used an equal-order velocity-pressure formulations to solve the Navier-Stokes equations in their three-dimensional mould filling simulation.

Rajupalem, Talwar and Friedl [92] developed a fully three-dimensional filling simulation tool. The discrete form of the pressure and velocity fields was of the same order, and therefore at once improving the accuracy of the pressure solution and eliminating the issue of spurious pressure modes. To solve the presence of false oscillations caused by the incorrect treatment of different time scales, both mesh and time-step sizes must be strictly reduced. This is impracticable in 3D flow situations, apart from perhaps very simple geometries.

Kim and Turng [93] conducted 3D numerical simulations for the injection moulding filling phase using different algorithms and finite element types. They introduced a simple memory management procedure to deal with the large sparse matrix system avoiding the need of a huge amount of storage space. The numerical simulation worked well in predicting the melt front position, for the cavity filling of a 3D optical lens, and was shown to be useful to guide the identification of better processing conditions.

Haagh and Vosse [94] employed a pseudo-concentration method in a FEM program for the filling stage of the injection moulding process. The proposed model has proven to give satisfactory results for the filling of both two– and three-dimensional moulds, and for the advancement of the flow front and the fountain flow effect. From these results, the authors considered the pseudo-concentration method as a promising technique for mould filling simulations, as long as the boundary conditions are properly defined.

Chang and Yang [91] developed a numerical algorithm for the mould filling stage of the injection moulding process based on an implicit finite volume approach (FVM). The numerical model deals with the 3D isothermal flow of incompressible, high-viscous Newtonian fluids with moving interfaces. To discretize and solve the Navier-Stokes equations the collocated FVM and the SIMPLE segregated algorithm were used. This method showed to be a very promising alternative to work with injection moulding problems, combining the FVM and the algebraic volume of fluid method (VOF). Comparing the 3D models with the Hele-Shaw predictions the results for the filling of thin cavities were similar. However, the proposed numerical model predicted with better efficiency and accuracy the cavity filling which has a larger thickness change as well as the filling of thick cavities. Additionally, it was possible to simulate the complex 3D mould filling process on a personal computer system due to the efficiency of the presented numerical algorithm in terms of CPU time and memory requirement.

There were also other similar methodologies to describe the 3D injection moulding process, as the ones proposed by Chang *et al.* [95], Zhou *et al.* [96] and Araújo *et al.* [97].

Chang *et al.* [95], developed a numerical model to simulate the mould-filling behaviour in the plastic encapsulation of microelectronics, using the same approach that the one used by Chang and Yang [91]. The simulated results agreed with experimental data which demonstrate the applicability of the model for practical plastic encapsulation simulations.

Zhou and Turng [96] presented a FVM hybrid flexible distributed/ shared memory parallel programming model to simulate the filling phase of the injection mould process, which is suitable to be used on computational clusters. In this study were conducted moulding experiments to be compared with the simulation results, the comparison showed good agreements between the numerical prediction and the experimental data. The results achieved from the studies tested on the computational clusters showed that the proposed implementation scales well, and, while the accuracy and stability were kept. Consequently, the computation time for complex geometry was greatly reduced. Therefore, optimization studies of the injection moulding process are possible due to the efficiency of the algorithm in terms of CPU time.

Araújo *et al.* [97] developed a parallelized algorithm for the simulation of non-isothermal 3D mould filling, where the proposed algorithm calculates the 3D velocity, pressure and temperature fields. Similarly to Chang and Yang [91], the multiphase flow was modelled resorting to a VOF based approach. The model was validated for moderate Reynolds number flows using a structured mesh and it was applied to highly viscous flow problems representative of those associated with the injection moulding process. Additionally, the model showed to have good accuracy and stability.

Estacio and Mangiavacchi [98] and Jiang *et al.* [99] proposed a mixed FVM and FEM to solve flow and heat transfer in injection moulding. This method combines the flexibility of FEM to discretize complex geometries with the conservative formulation of the FVM. Hence, this method takes advantage of the positive properties of both the FEMs and FVMs. Additionally, the FVM is more efficient in solution speed and memory usage, especially for large problems [100].

3. AVAILABLE SOFTWARE CREATED FOR THE INJECTION MOULDING PROCESS

The injection moulding process is an intricate, dynamic, and transient process, involving convoluted melting-flow-pressure-solidification phases, for which appropriate numerical tools can be a useful design tasks support. In the market are available some commercial software of which this state of art gives special prominence to Moldex3D® and MoldFlow® Insight.

Moldex3D® software is a product developed by Coretech System Co. incorporating the most advanced analytical technologies [101]. Moldex3D® has a complete set of simulation tools for plastic products injected with hybrid solid mesh technology and the high-performance FVM [102]. This program simulates the standard injection moulding process, but also has add-ons capable of simulating non-conventional injection moulding processes such as bi-injection, co-injection, powder injection moulding and others [101]. The standard process simulation includes the filling, packing and cooling phases of the injection moulding process, the simulations can be carried out considering the problem as 2D or 3D. In the case of the 3D simulation, it uses the Navier Stokes equations and considers, by default, that during the filling phase the fluid is incompressible and that during the packing phase it is compressible, to make the simulation faster [103]. However, it is also possible to define that throughout the process the fluid is compressible. This program offers several models for viscosity as Newtonian, power law, Cross-WLF and Carreau model [104] as also presents viscoelasticity models [105]. For the equation of state presents models as Spencer-Gilmore model, Tait model, Modified tait model and others [29].

MoldFlow® Insight software owned by Autodesk, Inc. simulates the standard injection moulding process and, similarly to Moldex3D®, has tools that can simulate non-conventional processes [106]. In case of the standard injection moulding process the program can simulate the filling, packing and cooling phases considering the problem as a 2D or a 3D [107]. In MoldFlow® Insight it is possible to choose between performing a fast fill analysis, in which the flow is considered incompressible and have an aggressive melt front advancement, or a standard fill analysis where is consider a compressible flow with a conservative melt front advancement and with a tight convergence criterion, thereby slowing the analysis down but getting accuracy in the results [108]. As can be very complex predicting how a molten material flows in a mould, MoldFlow® Insight presents several models which vary in the dependencies they consider and the moulding process they address [109].

These programs are widely used by the industry, but not all companies have the financial capacity to pay the high licenses these programs have, so began to appear open-source codes to replace the commercial ones, some examples are presented hereafter.

The openInjMoldSim [110, 111] solver is an injection moulding open-source solver based in OpenFOAM®, stands for Open source Field Operation and Manipulation, is a free and open-source CFD toolbox. OpenFOAM® is used in academia and industry to solve a wide variety of computational problems. Unlike to any proprietary software, it is possible to access and modify the source code. Taking this into consideration, Kristjan Krebelj and Janez Turk [110, 111] implemented the modified Tait equation of state and the Cross-WLF model in the *compressibleInterFoam* solver, of the OpenFOAM library and called *openInjMoldSim*. This modified solver aims to simulate the filling and packing phase of the injection moulding process, considering the compressible flow, and was validated on a 2D demo test case for an amorphous polystyrene grade, the simulation with semicrystalline polymers is under development [110, 111].

Also based in OpenFOAM®, Felix Ospald [112] implemented the Carreau-WLF model in the *interFoam* solver, for the simulation of injection moulding with short fibre reinforced thermoplastics in a laminar flow regime. In this solver, the flow is considered incompressible and only simulates the filling phase. The filling results were in good agreement with the results of MoldFlow®.

Another example of an open-source tool is the work done by Billy Araújo [113] who developed a solver to simulate the 2D injection moulding process and also developed a parallel 3D unstructured non-isothermal flow solver for the simulation of the injection moulding process. In this solver, it is considered that air and polymer behave like incompressible fluids during the filling phase and that both were approximate as a generalized Newtonian fluid. The results obtained showed good precision and a reasonable parallel efficiency and scalability [97].

Comparing the commercial tools with the open-source tools is remarkable the difference between both, while the commercial tools are very developed and consider all phases of the injection moulding process (filling, packing and cooling) the open-source tools are still only able to predict the filling and packing phase, in the case of packing phase only for 2D cases and just for amorphous polymers. Another difference is in the models that commercial software has implemented and that in the case of the open-source programs only have one model implemented.

CHAPTER IV – CASE STUDY

1. INTRODUCTION

To improve performance and reduce resources consumption, the use of appropriate modelling tools is mandatory. There are very complete commercial programs on the market capable of simulating the injection moulding process but associated with these programs are the costs of licenses that not all companies are capable to support. These costs can be overcome by using the open-source modelling tools, but these are still far beyond the potential that commercial programs can offer. Also these tools are not fully validated and therefore their use in the industry is not yet viable/ possible. Thus, the main objective of this chapter is to present a case study, where the openInjMoldSim [110] solver (an injection moulding open-source solver based in OpenFOAM® computational library) is validated by comparison against the commercial software Moldex3D® [101]. For this case, the version used was Moldex3D® R15 SP1OR and *OpenFOAM-5.x* version.

2. METHODOLOGY

For the accomplishment of this study, a methodology was followed in order to compare both programs and thus to validate the openInjMoldSim. First, the geometry to be studied was defined, then the models that both programs used to simulate the injection moulding process were presented. Subsequently, three meshes were generated starting at a coarser to a more refined mesh. After the mesh generation, it was necessary to define all the initial and boundary conditions for the flow under study in order to be able to run the simulation.

This methodology is presented in the next subsections.

2.1 GEOMETRY

This case study is related to the filling of a rectangular cavity with a cylindrical insert, illustrated in Figure 12. The rectangular cavity has a constant thickness of 1 mm, length of 30 mm and width of 4 mm, the cylindrical insert has a diameter of 2 mm. In literature, this type of cavity with simple inserts is common [114] and the one used in this work has the same dimensions of the part presented in Zhuang *et al.* [115]. The behaviour of the polymer melt around a cylinder comprises flow separation, weld line, flow cross-section modification, thus provides useful insights in understanding the flow in more complex geometries. Additionally, the part has a small thickness when compared to the other dimensions, which is a typical characteristic of the injected moulded parts, which further justifies the option for this geometry.

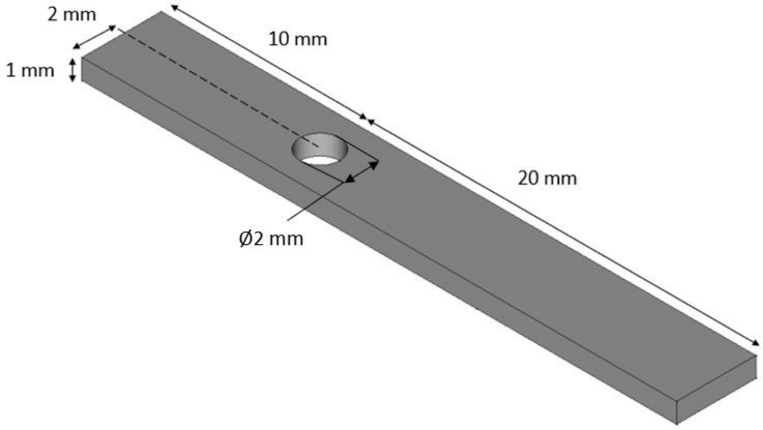

Figure 12 – Geometry and dimensions of the part under study.

To be able to run the simulation in any numerical code, the problem boundary conditions (patches in OpenFOAM® terminology) must be defined in all the faces, which for this case study are shown in Figure 13. The entrance of the material is done through the Inlet face and the Outlet is the face where the air leaves the system. All the other faces are as impermeable walls where only heat exchange between mould and material can occur.

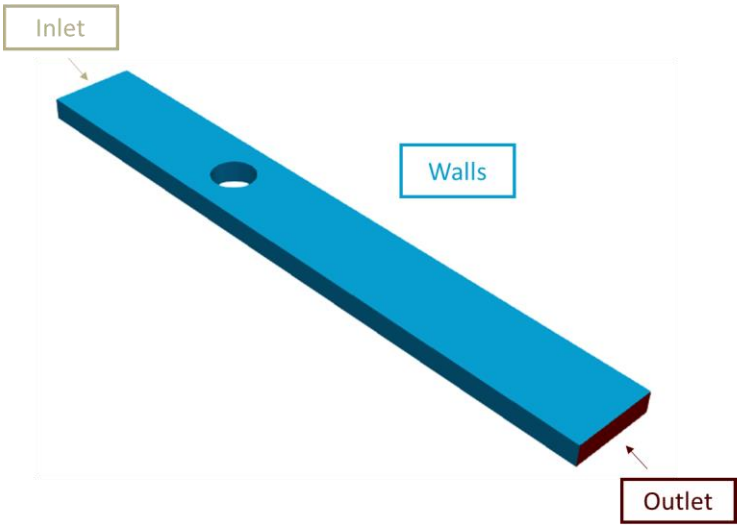

Figure 13 – Case study boundary surface groups.

2.2 MODELS USED IN THE SIMULATION

The injection moulding process was modelled with the Navier-Stokes equations for compressible fluid flow (Equations (29), (31) and (35)), using the Cross-WLF model (Equations (9), (11) and (12)) and the Modified Tait-equation (Equations (23 – 28)) as the viscosity model and Equation of State, respectively. The FVM was used to numerically discretize the equations in each domain control volume.

As mentioned in the state of the art, the solver developed by Kristjan Krebelj and Janez Turk [110] used in this work, considers that the flow is always compressible, i.e., takes into account the compressibility of the material during the filling and packing phase [111], contrary to almost all authors which ignore the compressibility of the polymers during the filling phase [103, 104].

Following the VOF approach, the evolution of the melt front is calculated by the following transport equation

$$\frac{\partial \alpha}{\partial t} + \mathbf{u} \cdot \nabla \alpha = 0 \tag{43}$$

where α is a concentration function taking the value of the volume fraction of the fluid in a domain control volume. When α is equal to 1 it means that the control volume is full of polymer, and when it takes the value 0 the control volume is full of air (see Figure 14). The melt front is located within cells with concentration values between 0 and 1, as shown in Figure 14.

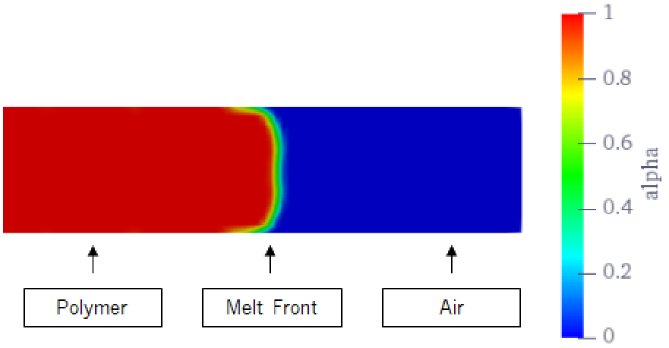

Figure 14 – Phase indicator α distribution for a typical injection moulding case study.

With the VOF approach, the calculation domain is treated as a one-phase system, which properties are averaged between the ones of the polymer melt and air, weighted with the phase indicator (α) value. Thus, in governing flow equations (Equations (29), (31) and (35)), the physical parameters are expressed using the mixture law weighed by α. For example, the density, ρ, is computed as

$$\rho = \rho_{poly}\alpha + \rho_{air}(1-\alpha) \tag{44}$$

where ρ_{poly} and ρ_{air} are the polymer and air densities, respectively. Using this type of expression for the physical parameters involved in the flow equations, it is possible to solve the flow problem for the entire domain [118].

2.3 MATERIAL

The selected material to be used in the numerical studies performed in this thesis was the General Purpose Polystyrene (GPPS) – Styron 678 Americas Styrenics, which is a widely employed injection moulding grade and, considering the possible experimental studies, is available in the Department of Polymer Engineering from the University of Minho. GPPS is a transparent polystyrene, amorphous, rigid, low-cost material which is easy to process and is one of the largest volume commodity thermoplastic [119].

The recommended processing conditions and thermo-physical properties of the GPPS were obtained from the MoldFlow® software database, and are presented in Table 5. Thermophysical properties such as heat capacity (c_p), thermal conductivity (k), density (ρ), maximum shear rate ($\dot{\gamma}_{\max}$) and maximum shear stress ($\sigma_{\max}$) at which the material can be subjected are presented, as well presents the recommended conditions for processing the GPPS like the temperature range at which the material should be injected (T_{melt}) and at which the mould should be (T_{mould}). Table 5 also shows the temperature at which the material solidifies (T_{freeze}) and the temperature at which the part to be moulded must be ejected ($T_{ejection}$). The Cross-WLF model parameters are shown in Table 6 and the modified Tait-Equation parameters in Table 7, also obtained from the MoldFlow® software database.

Table 5 – Thermo-physical properties and recommended conditions to process GPPS, based on MoldFlow® database.

c_p	2100	J/(kg.K)	T_{melt}	180 – 280	°C
k	0,15	W/(m.K)	T_{mould}	20 – 70	°C
ρ	1035,9	kg/m^3	T_{freeze}	101	°C
$\dot{\gamma}_{\max}$	40000	1/s	$T_{ejection}$	80	°C
$\sigma_{\max}$	2,4x x10^5	Pa			

Table 6 – Cross-WLF model parameters for GPPS – Styron 678 Americas Styrenics, based on MoldFlow® database.

Cross-WLF model	n	0,2903	
	τ^*	13678	Pa
	D_1	7,44x10^{10}	Pa.s
	D_2	373,15	K
	D_3	0	K/Pa
	A_1	25,971	
	A_2	51,6	K

Table 7 – Modified Tait-equation parameters for GPPS – Styron 678 Americas Styrenics, based on MoldFlow® database.

Modified Tait-Equation	b_{1L}	0,0009881	m^3/kg	b_{4S}	0,003106	1/K
	b_{2L}	$7,03x10^{-7}$	$m^3/(kg.K)$	b_5	373,98	K
	b_{3L}	$1,71x10^{8}$	Pa	b_6	$2,88\ x10^{-7}$	K/Pa
	b_{4L}	0,004495	1/K	b_7	0	m^3/kg
	b_{1S}	0,0009873	m^3/kg	b_8	0	1/K
	b_{2S}	$2,89\ x10^{-7}$	$m^3/(kg.K)$	b_9	0	1/Pa
	b_{3S}	$2,43\ x10^{8}$	Pa			

2.4 MESH GENERATION

To be able to compare the results obtained with Moldex3D® against the ones obtained with openInjMoldSim solver, the ideal scenario would be to make the computations with the same meshes. However, in Moldex3D® the mesh is generated in the program itself and only one type of mesh can be generated, the boundary layer mesh (BLM), this is a hybrid mesh that contains two kinds of elements, prismatic elements of reduced thickness on the surface and tetrahedral elements inside [120]. While in OpenFOAM® the *cfMesh* utility is used to generate a cartesian mesh, which is able to reduce the non-orthogonality of the mesh, and, therefore, makes easier to solve the flow equations due to the simplifications obtained when discretizing the diffusive terms.

As it was not possible to generate the same type of meshes in both software, the objective was then to generate meshes with, approximatively, the same number of computational cells. For each software were generated three representative meshes, where the number of divisions along each edge is doubled for each two consecutive refinements levels. As it is a 3D simulation the number of cells between each refinement level is approximately multiplied by 8. The number of cells in each mesh in both programs is shown in Table 8. The meshes were obtained by a trial and error procedure until the number of cells between the meshes was as close as possible, always maintaining the quality of the mesh.

Table 8 – Number of cells in each mesh, in Moldex3D® and openInjMoldSim software.

	Moldex 3D	**openInjMoldSim**
Mesh 1	19509	19852
Mesh 2	158322	180600
Mesh 3	1297509	1337276

Figure 15 illustrates Mesh 1 for both software, and it is possible to observe the differences in the type of meshes generated, namely on the shape of the elements obtained. The number of cells in Moldex3D appears to be much smaller than the number of cells in openInjMoldSim, but by the Table 8 this difference is not so noticeable, the number of cells in Moldex3D® is given by the program, but it is not known how the program counts them.

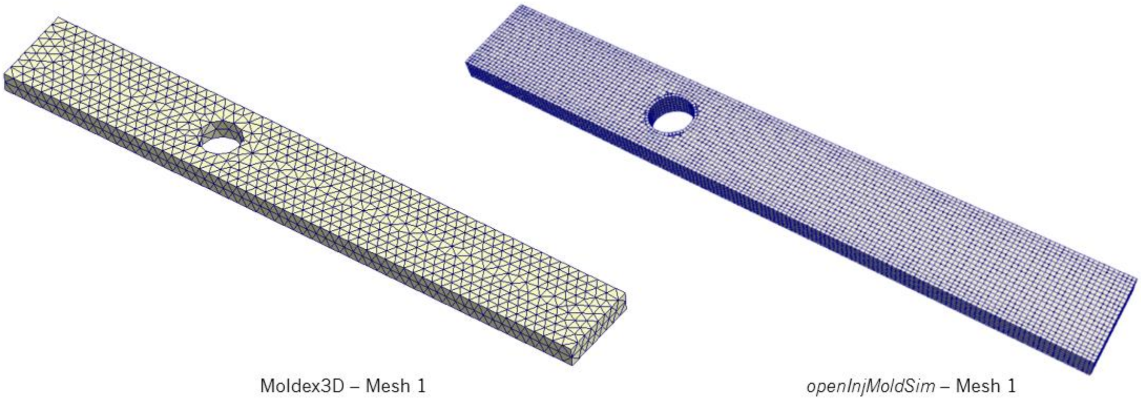

Figure 15 – Mesh 1 from Moldex3D® and openInjMoldSim.

2.5 INITIAL AND BOUNDARY CONDITIONS

For the filling phase, the initial and boundary conditions were defined considering the properties of the polymer selected, GPPS, and the range of values defined by the material technical data sheet (see

Table 9).

Table 9 gives the initial conditions that were set on Moldex3D®. The filling time was set considering the suggestion made by the program Moldex3D®, which directly imposes the flow rate at the inlet.

Table 9 – Initial conditions used for the filling phase simulation with Moldex3D®.

Initial conditions of the filling phase on Moldex3D®	Filling time	0,1	s
	T_{melt}	230	°C
	T_{mould}	50	°C
	$T_{ejection}$	80	°C

When using *openInjMoldSim* it is not possible to define the filling time, but instead it is possible to impose either a velocity or a pressure profile at the Inlet. Therefore, first a simulation was run in Moldex3D® and the flow rate was obtained as a function of the filling time, see Figure 16. Subsequently, the flow rate obtained was divided by the inlet cross-section area, thus obtaining the evolution of the average velocity at the same location, when using the openInjMoldSim solver.

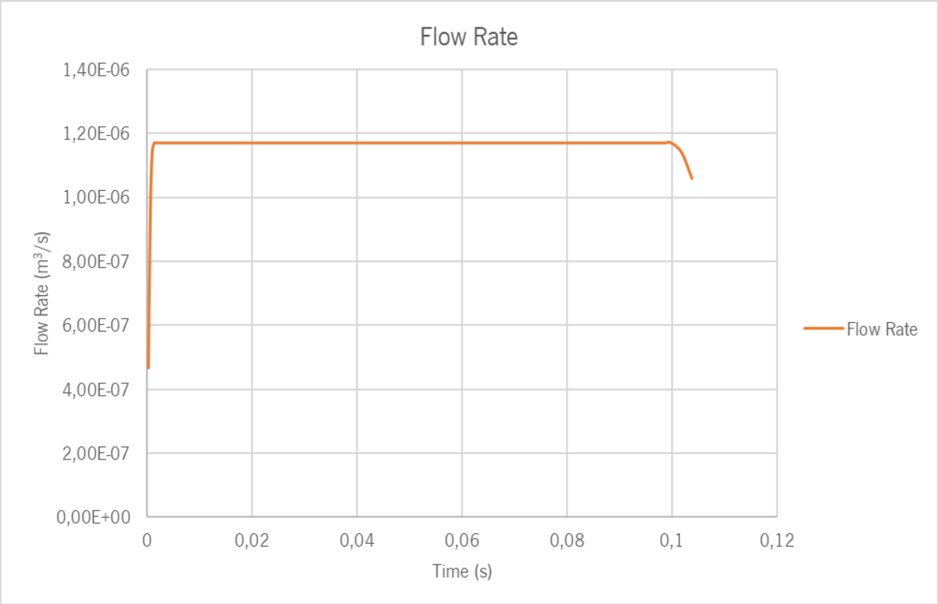

Figure 16 – Flow rate vs time obtained from the simulation result in Mesh 1 using Moldex3D®.

The initial and boundary conditions for each field, temperature (T), pressure (p), velocity ($\boldsymbol{u}$), shear rate ($\dot{\gamma}$) and concentration function (α), need to be defined when using the openInjMoldSim solver. Notice the same values are required for Moldex3D® but the software does not require that input from the user, thus in many cases it is unclear the options adopted by Moldex3D®.

Figure 17 illustrates the initial boundary conditions for the temperature field. The temperature defined at the Inlet, is a fixed temperature of T = 230°C. At the Walls heat flux was considered so it was imposed the boundary condition externalHeatFluxTemperature, which is already implemented in the openInjMoldSim solver. For this boundary condition to be fully defined it is necessary to introduce the mould temperature (wall temperature, T_w) and the heat transfer coefficient (h), T_w = 50 °C and h = 1250 W/(m^{2}K) respectively. At the Outlet the boundary condition zeroGradient is employed, which imposes a fully developed boundary, thus it extrapolates the cell centre temperature of the boundary adjacent cell to the outlet face. The temperature inside the mould, which means the temperature of the air was considered to be equal to the temperature of the polymer, so the initial condition is T = 230 °C.

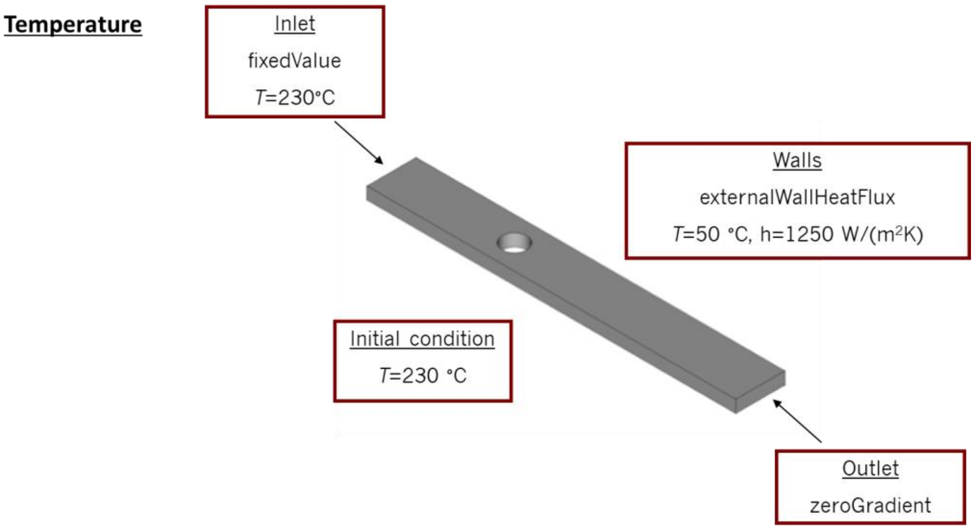

Figure 17 – Initial and boundary temperature conditions.

Figure 18 illustrates the initial and boundary conditions of the velocity field. The initial velocity imposed at the internal cells was $\boldsymbol{u}$ = (0 0 0) m/s. At the Inlet, the average velocity values were those obtained through the approach described above, in order to apply the same conditions in both programs. At the patch Walls there is no movement, therefore, the no-slip velocity is imposed, i.e., it takes a fixed value of $\boldsymbol{u}$ = (0 0 0) m/s. During filling, the air leaves through the Outlet, which means that a fully developed velocity profile near the outlet is assumed and, therefore, the zeroGradient boundary condition can be imposed.

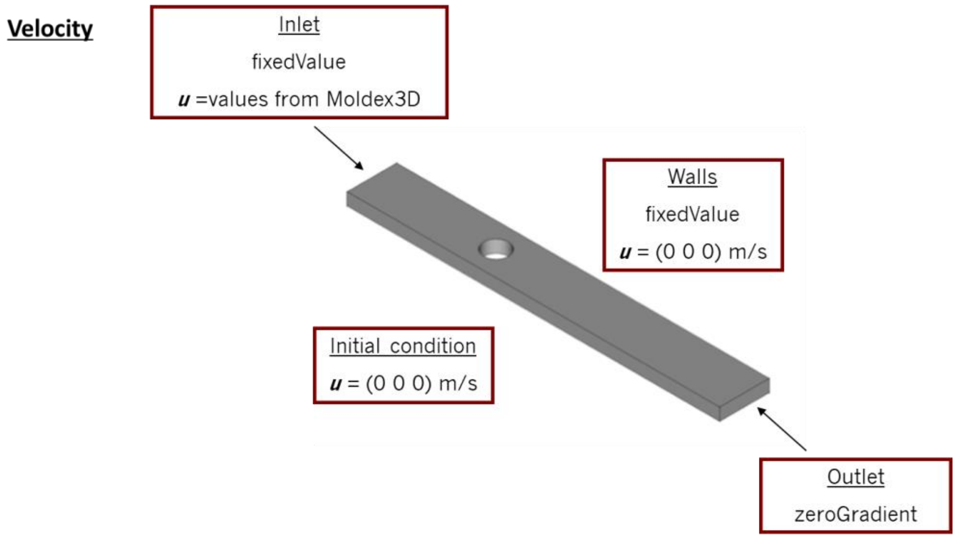

Figure 18 – Initial and boundary velocity conditions.

Figure 19 illustrates the initial and boundary conditions of the pressure field. Initially, the cavity is at atmospheric pressure p =1×10^5 Pa. The Outlet is considered to be at atmospheric pressure. Since a velocity value has been set at the Inlet, the pressure cannot be imposed, therefore the zeroGradient boundary condition is considered at that location. The boundary condition for the patch Walls is calculated, this is because the pressure value on the Walls will depend on other field quantities.

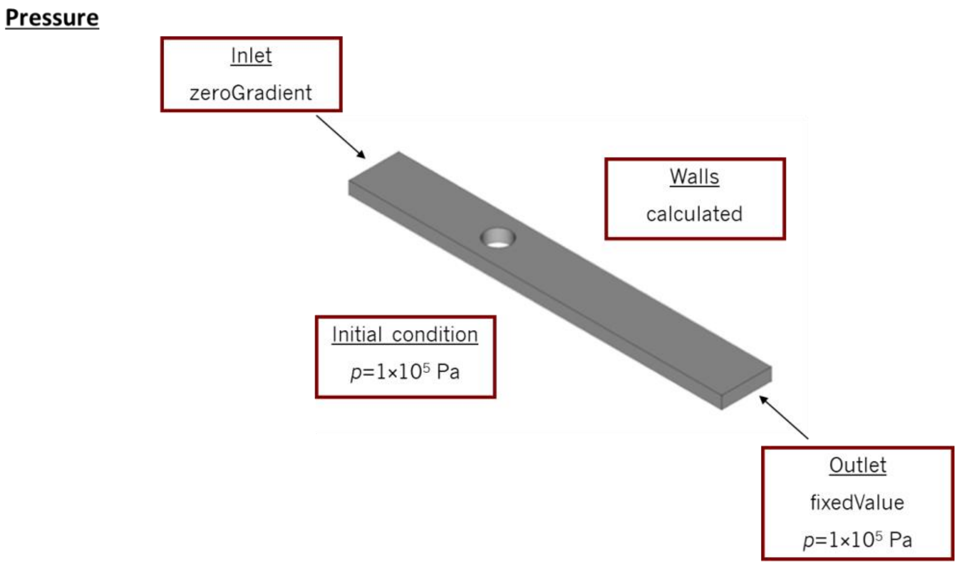

Figure 19 – Initial and boundary pressure conditions.

Figure 20 illustrates the initial and boundary conditions of the shear-rate field. All conditions are imposed as calculated, because the value of the shear rate depends on the velocity field (see Equation 5)

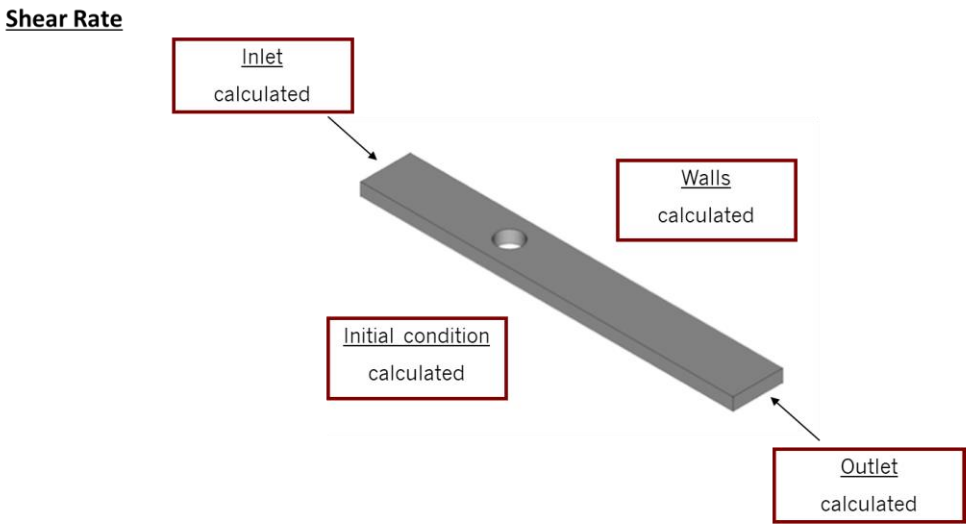

Figure 20 – Initial and boundary shear rate conditions.

Finally, Figure 21 illustrates the initial and boundary conditions of the concentration field (see Equation 38). Initially, inside the cavity there is just air, therefore α=0. At the Inlet, where the molten polymer enters the cavity, the fixed value α=1 of the concentration function is imposed. At the Walls and Outlet boundaries the zeroGradient condition is imposed, thus assuming an equal value to the one of the value of the nearby cell.

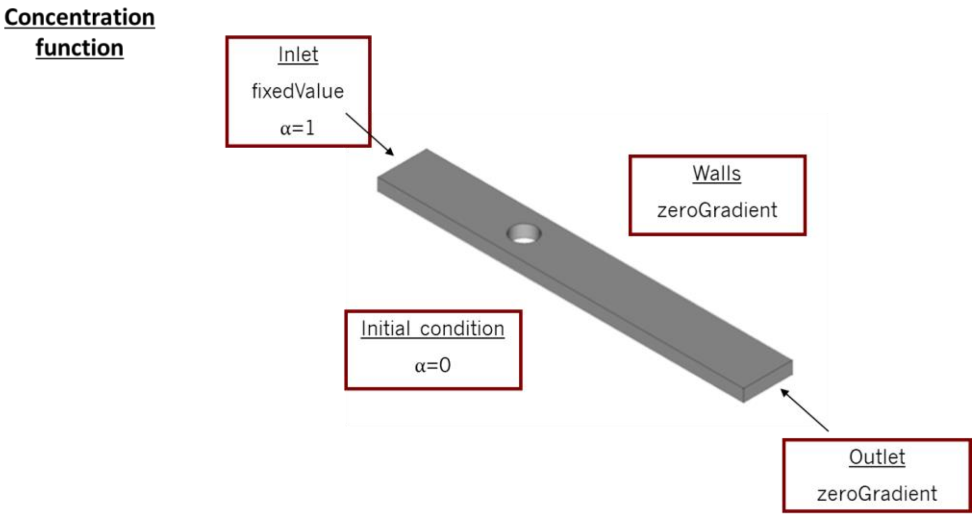

Figure 21 – Initial and boundary concentration function conditions.

Additionally, as *openInjMoldSim* solver does not have a material database as Moldex3D®, the material physical properties were defined manually in the program, both for the air and the polymer melt. The parameters of the modified Tait and Cross-WLF equations and the heat capacity (C_p), thermal conductivity (k) and the temperature at which the material solidifies (T_{freeze}) need to be specified. In Moldex3D® the air properties are not accessible to the user. In openInjMoldSim air was considered as a pseudo-fluid with a viscosity of η=0,1 Pa.s, which is a high value, usually employed to improve the convergence of the volume-of-fluid type algorithm. The convergence improvement is due to the reduction of the viscosity gradient at the interface between the two phases, as the mixture viscosity is computed as a linear interpolation of both polymer and air viscosities.

3. RESULTS AND DISCUSSION

With all the conditions set, the simulations were performed in both programs. Table 10 lists the execution times for each mesh in both programs, the difference between the simulation time of the two programs is significant being much faster in Moldex3D®, on average 18x faster. Assuming that both programs use a similar formulation may raise some issues, such as the meshes do not have the same degree of refinement in both programs. After obtaining the results, they were analysed and compared starting from the flow front and then analysing and comparing the temperature, velocity and pressure profiles. In the next results obtained will be presented and discussed.

Table 10 – Simulation time for each mesh in both programs.

	Moldex 3D	**openInjMoldSim**
Mesh 1	2 minutes	43 minutes
Mesh 2	28 minutes	392 minutes
Mesh 3	300 minutes	5328 minutes

3.1 MELT FLOW FRONT

The flow front was the first result to be analysed. As the numerical method used was the FVM mass conservation is always assured [35], so when the same flow rate is imposed in both programs the flow front must be at a similar position, i.e. the volume of material at each instant of time should be the same in both programs.

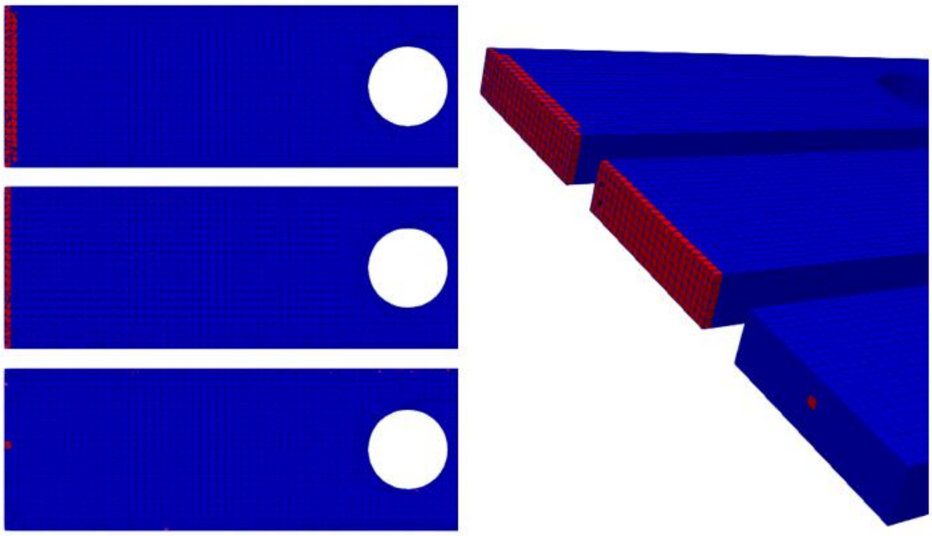

Figure 22 – Tests made for the amount of the initial filled cavity, in the case of Mesh 1. The polymer and air inside the cavity are represented by red and blue, respectively.

There is a factor that has not been mentioned yet, with the conditions defined previously, the simulation in openInjMoldSim did not run when the cavity was empty, because it requires some material inside the mould at the beginning of the simulation. Thus, in order to accomplish this requirement, it was initially defined that the concentration function would be 1 in two rows of cells, and this amount of material was gradually reduced, so that similar conditions to the ones of Moldex3D® are obtained. Figure 22 shows the reduction of the initial material inside the cavity in the case for Mesh 1, starting with two rows of filled cells, decreasing to a single row and lastly only with one filled cell.

To verify the similarity of the conditions, in openInjMoldSim the instantaneous flow rate was computed as given by

$$\dot{\alpha} = \frac{dV_{fraction}}{dt} \times V_{part} \tag{45}$$

Where $V_{fraction}$ is the volume fraction, t is time and V_{part} is the volume of the part under study.

In the graphic, shown in Figure 23, the evolution of the instantaneous flow rate over can be observed for the different attempts of the initial quantity of material inside the cavity, in the simulations run with the openInjMoldSim solver. As expected, with the decrease of the initial quantity of material inside the cavity, there was an approximation of the openInjMoldSim results to the Moldex3D® predictions.

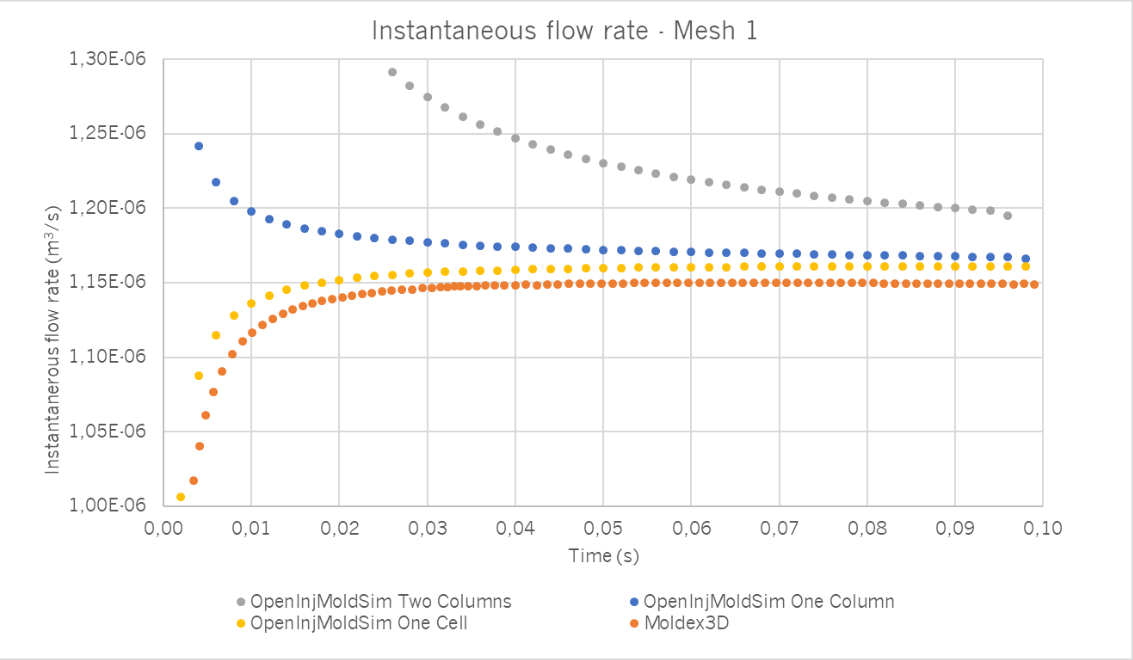

Figure 23 – Evolution of the flow rate over time for the different attempts of initial quantity of material inside the cavity in the simulations of the openInjMoldSim and for Moldex3D®.

In the case of Mesh 2 and 3 the same approach was used, reducing the initial amount of material to a minimal amount, that assured that the simulation was running in openInjMoldSim, but also ensuring that the conditions in both software are as close as possible. The evolution of the flow rate for Mesh 2 and 3 are presented, respectively in Figures 24 and 25.

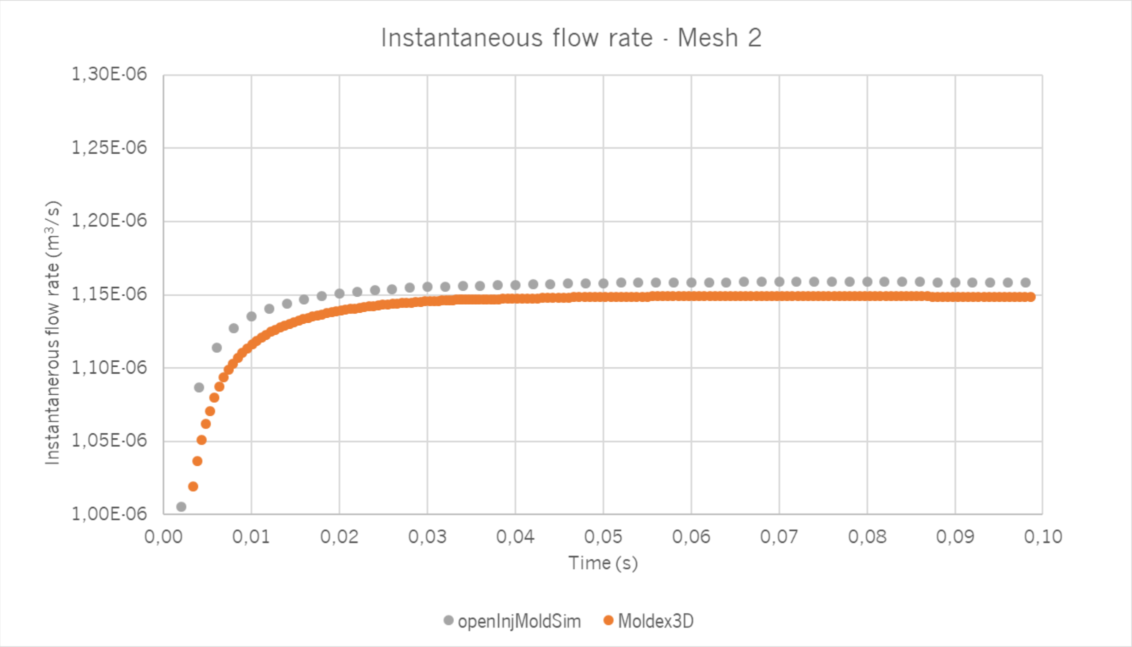

Figure 24 – Evolution of flow rate over time for Mesh 2 for openInjMoldSim and Moldex3D simulations.

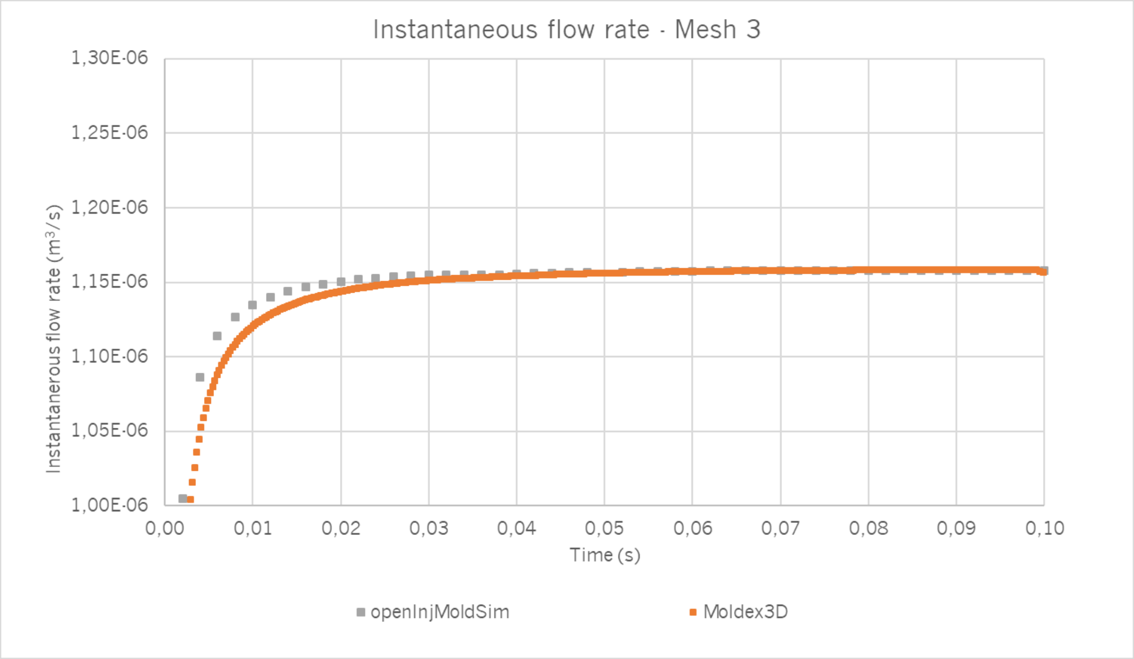

Figure 25 – Evolution of flow rate over time for Mesh 3 for openInjMoldSim and Moldex3D simulations.

By analysing the melt flow front in both programs, for Mesh 1, presented in Figure 26, there are slight differences between them in openInjMoldSim the melt flow front appears to be more advanced than in Moldex3D®. These differences may be related to the different type of meshes in both programs, the low level of mesh refinement and due to the initial filled cells required by openInjMoldSim. Nevertheless, by observing the melt flow front it is possible to see that the position and time of the welding line in both programs is rather similar.

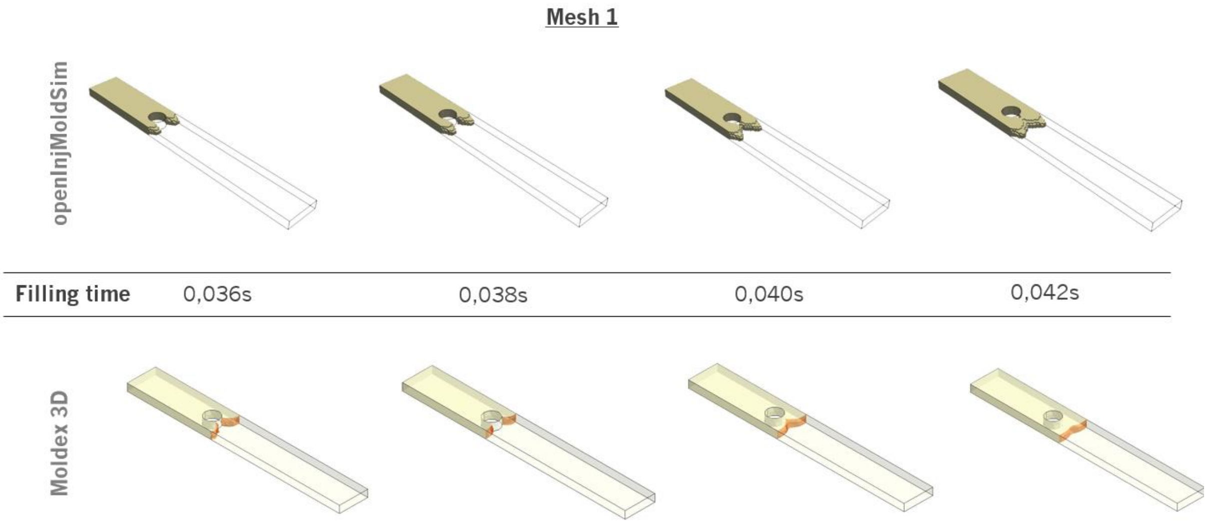

Figure 26 – Melt flow front evolution for Mesh 1 at 0,036s, 0,038s, 0,040s and 0,042s of filling time, for both programs.

For Mesh 2, when analysing the melt flow front in the two programs there is an approximation of the melt flow front behaviour (see Figure 27) in case of openInjMoldSim the melt flow front is smoother, thus getting closer to the results of Moldex3D®. It is also possible to observe that the welding line positions is the same in the two programs, similarly to what happened in the case of Mesh 1.

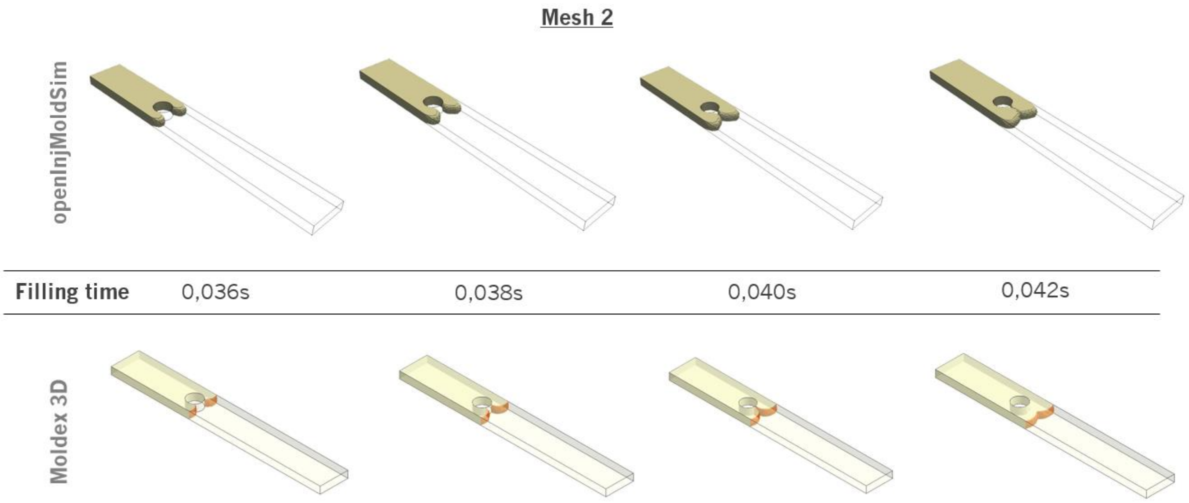

Figure 27 – Melt flow front evolution for Mesh 2 at 0,036s, 0,038s, 0,040s and 0,042s of filling time, for both programs.

The Mesh 3, being a more refined mesh, presents more cells in height and in the transversal direction which may promote the appearance of the expected parabolic flow front [7, 105]. In the case of openInjMoldSim, this is verified but in the case of Moldex3D® even with the refinement of the mesh this always presents a sharpest (almost vertical) melt flow front which is a strange behaviour.

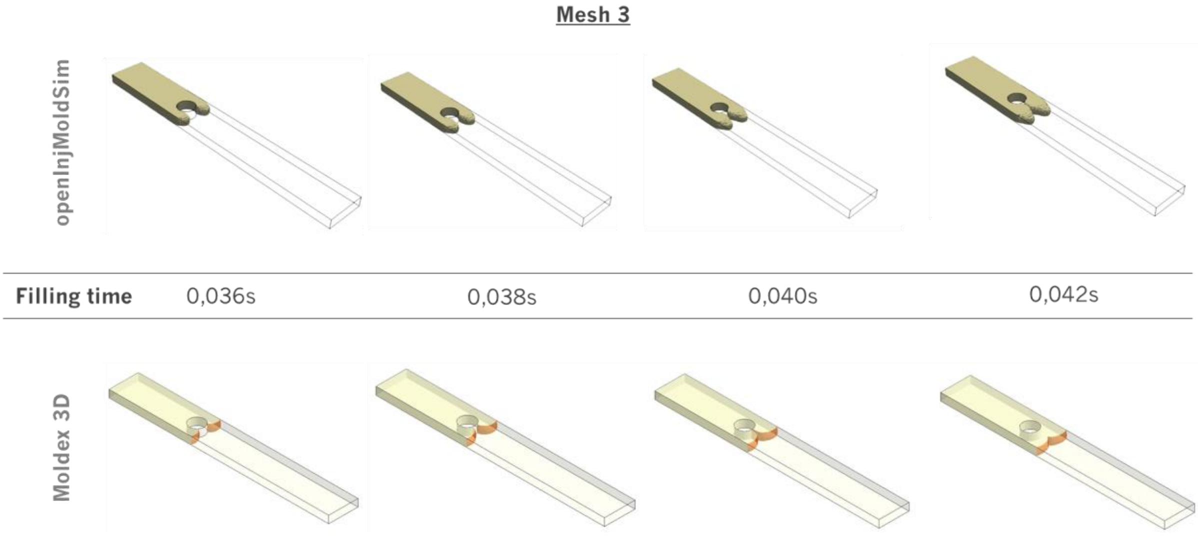

Figure 28 – Melt flow front evolution for Mesh 3 at 0,036s, 0,038s, 0,040s and 0,042s of filling time, for both programs.

While in openInjMoldSim the observation of the position of the welding line is only observable when analysing the flow front, Moldex3D® also presents a tool that allows determining the location, as well the temperature at which the flow front meet. This data might be relevant to evaluate the severity of the welding line formed, since the higher the temperature typically indicates a less severe weld line. This happens because the melt flow fronts will entangle much more efficiently for high temperatures, which helps to reduce both the visual appearance as well as potential mechanical weakness. On the other side, when the temperature is lower the melt will be more viscous and so it will be more difficult to mix the independent flow fronts, with the consequent problems for the appearance and mechanical behaviour.

In Figure 29, is shown the position and the temperature at which the melt flow front meet, as predicted by Moldex3D®. The lowest temperature value found is about 226 °C which is a high temperature, since the injection temperature was 230 °C.

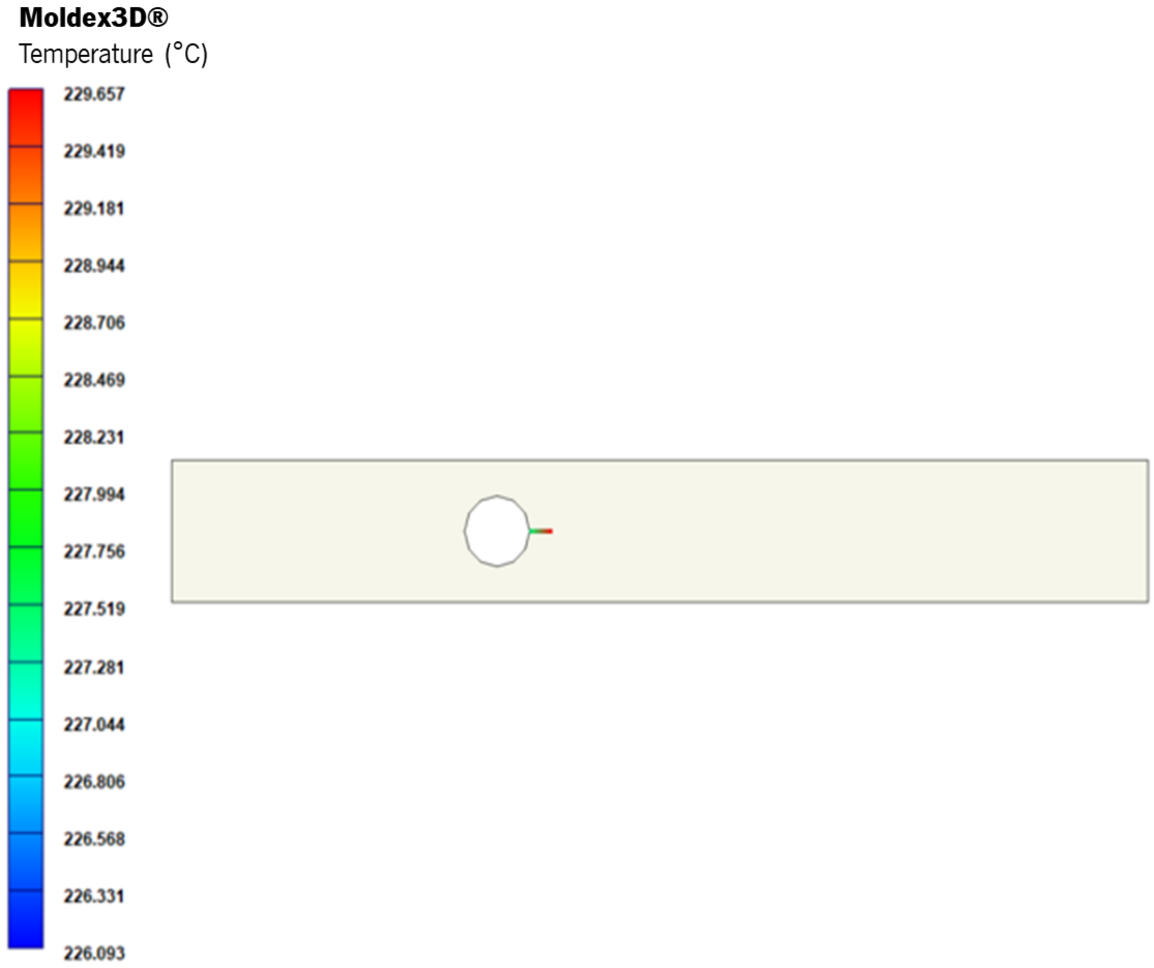

Figure 29 – Welding line position and temperature at which the melt flow front meet in Moldex3D®.

3.2 ANALYSIS OF TEMPERATURE PREDICTIONS

For additional comparison purposes, the temperature distribution was analysed along the line shown in Figure 30, the analysis was performed in this position because it is a critical one, after the melt flow front separation. In the case of Moldex3D®, in order to be able to analyse the results in a specific position, it is necessary to define, before running the simulation, points in the position in which the results will be analysed, because only this way the information about that specific position will be recorded. Thus, along the line shown in Figure 30, 42 points were defined. In the case of openInjMoldSim, this pre-processing is not necessary since all the information is stored along the time steps.

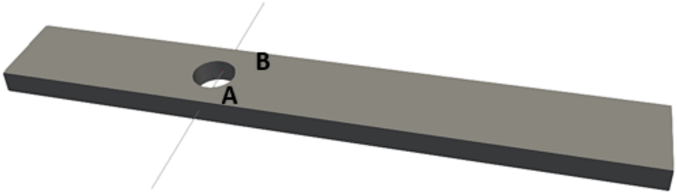

Figure 30 – Position of the line where the results were analysed.

In Figure 31 the temperature results, at 0,076s of filling time for the Mesh 1, are presented for both programs. The temperature distribution predicted by both software is significantly different. Additionally, the temperatures near the mould wall, in the case of Moldex3D®, are very low when compared to the temperatures obtained in openInjMoldSim. This observation required a more detailed review of the conditions imposed on Moldex3D®.

The simplifications that are made in Moldex3D®, to make the program easy to use, difficult the knowledge of the actual conditions employed, which in many cases are automatically defined by the software. Thus, there are assumptions that are only identified after identifying strange results, such as the ones of the case being analysed. In Moldex3D®, only the mould temperature (T_{mould} = 50 °C) was being imposed, but by the results obtained the temperature near the walls ($T_{wall} \approx$ 80 °C) is higher than the mould temperature, which means that a heat flux was also being imposed in the walls. The results shown in Figure 31, evidence that the heat flux imposed was not the same in both programs, due to the large temperature difference observed.

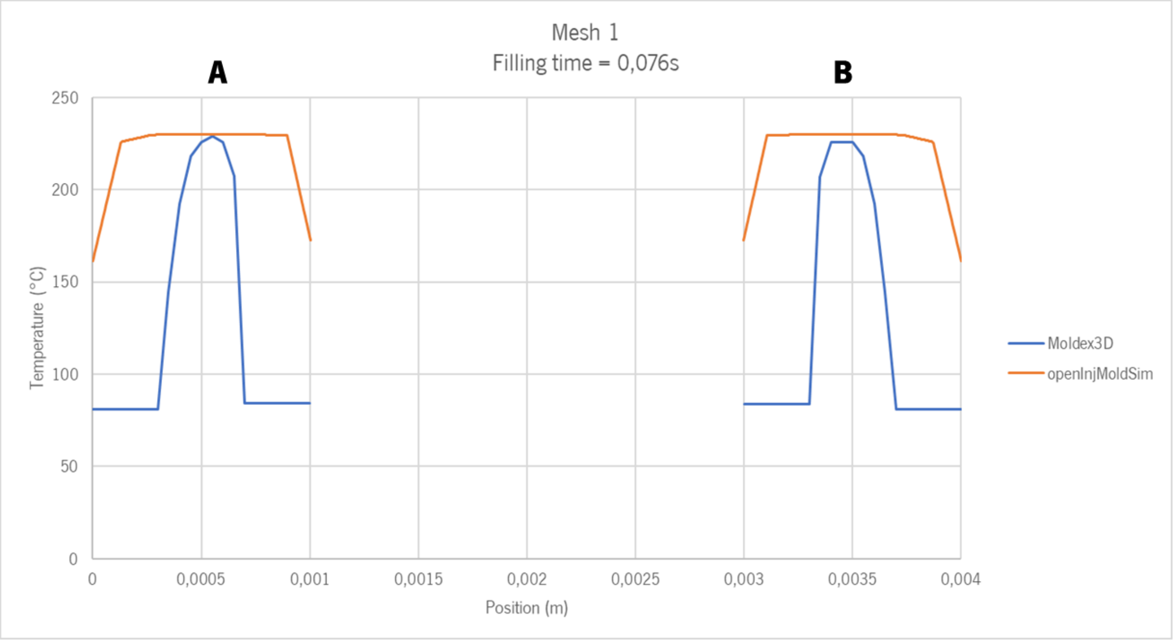

Figure 31 – Temperature distribution of openInjMoldSim and Moldex3D® program at 0,076s of filling time for the Mesh 1.

After a detailed analysis of the commercial program it was found that, by default, Moldex3D® defines an automatically determined heat transfer coefficient which means that based on process condition the heat transfer coefficient is calculated, but the value defined is not given to the user. In addition to the default setting, it is possible to set a uniform heat transfer coefficient, as desired in the current study. Thus, and like the condition defined in openInjMoldSim, the heat transfer coefficient was set as h = 1250 W/(m^{2}K).

The results of the new simulation are shown in Figure 32, the temperature next to the walls in Moldex3D® results approached to the temperature of the openInjMoldSim. However, even with similar conditions in both programs the temperatures profiles between the programs are very different, in the case of Moldex3D® next to the walls appear constant temperature levels. Physically, this result does not make sense, because when a heat flux is imposed it is expected that near the walls there would be a variation of temperature, just as in openInjMoldSim.

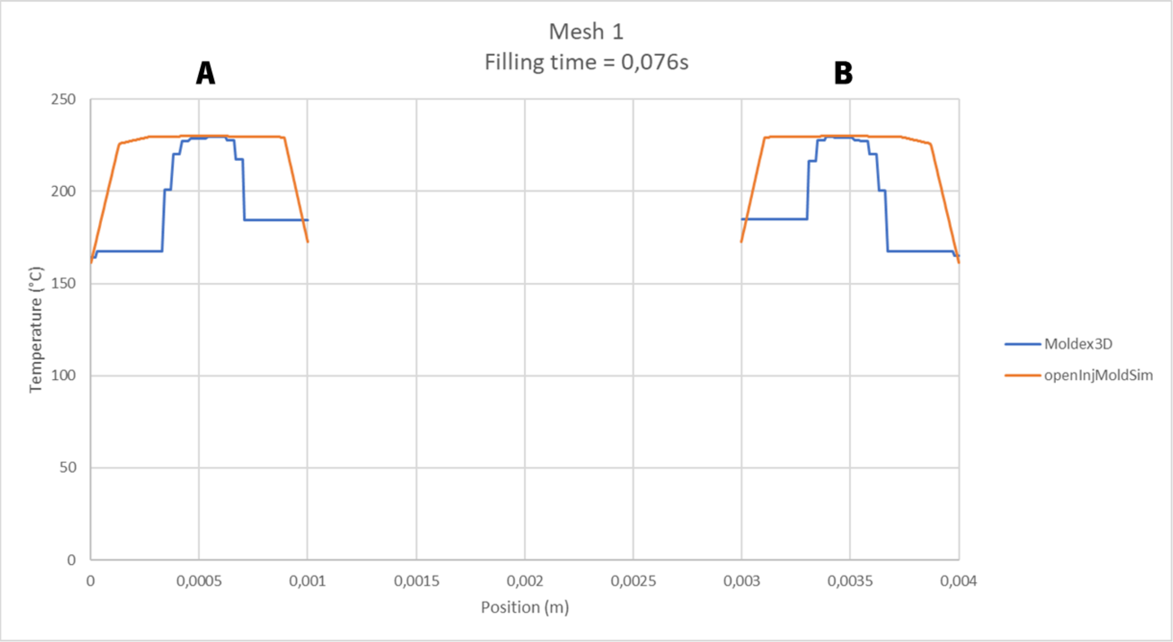

Figure 32 – Temperature predictions by openInjMoldSim and Moldex3D® programs at 0,076s of filling time for the Mesh 1, with the new heat transfer coefficient condition in Moldex3D®.

As previously commented, in Moldex3D® points are defined along the line under study. Since in this case the mesh is very coarse, several points are inside the same cell (see Figure 33) and depending on the way the program does the calculation, it can assume that all the points have the same value, thus justifying the existence of constant temperature levels next to the walls.

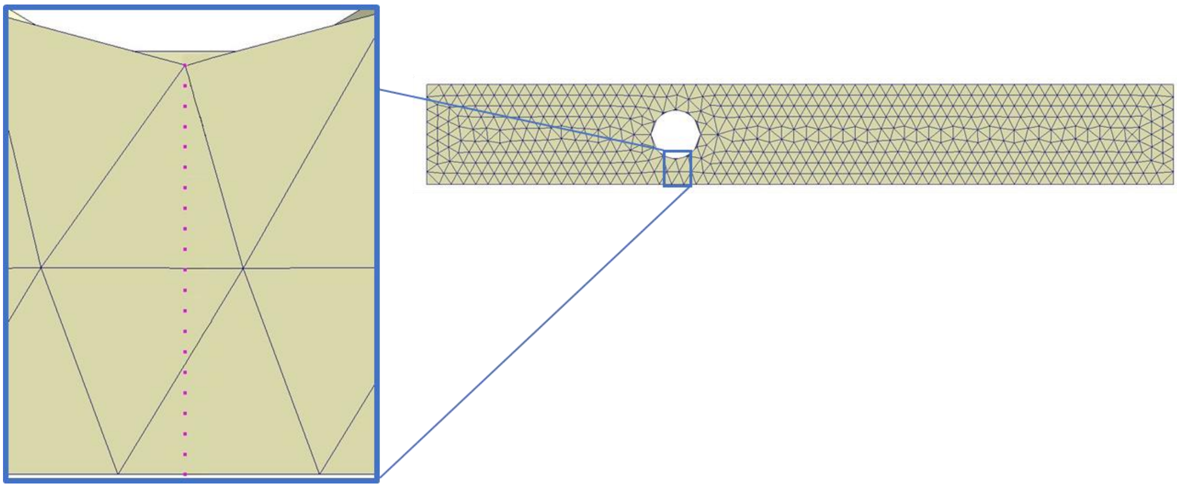

Figure 33 – Points taken along the line where the results were analysed for Mesh 1 in Moldex3D®.

The results obtained with Mesh 2 (Figure 34) are much more similar between the two programs. In the case of Moldex3D®, the plateau next to the walls has decreased, which allowed to conclude that the verified problem is associated with the level of mesh refinement. By refining the mesh, the size of the cells decreased, so the number of points inside the same cell decreases, also decreasing the size of the plateau. In case of openInjMoldSim, the temperature near the cylinder walls increased because with the refinement of the mesh the resolution increased, being more noticeable the generation of heat by viscous dissipation.

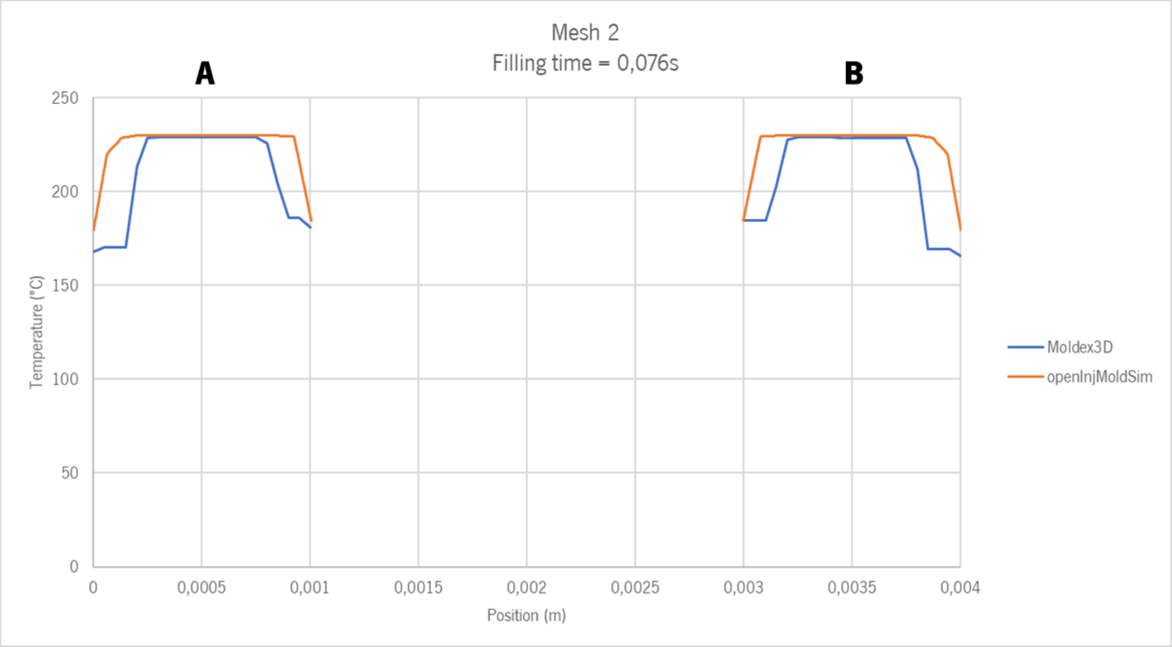

Figure 34 – Temperature predictions by openInjMoldSim and Moldex3D® program at 0,076s of filling time for the Mesh 2.

As for Mesh 2 there was a decrease in the constant temperature plateau next to the wall in the case of Moldex3D®, also for Mesh 3 there was a reduction of the same (see Figure 35). The temperature next to the cylinder in both programs increases with respect to the results of the previous mesh. The reason why it is verified is due to the higher resolution of the mesh making the viscous dissipation more noticeable.

In all meshes it was possible to observe that next to the inner cylinder the temperature was higher than the one at the external walls. The material next to the external walls is cooling along the whole channel, in case the material that goes in the centre of the channel only cools more quickly when it comes into contact with the cylinder insert, hence the temperature difference.

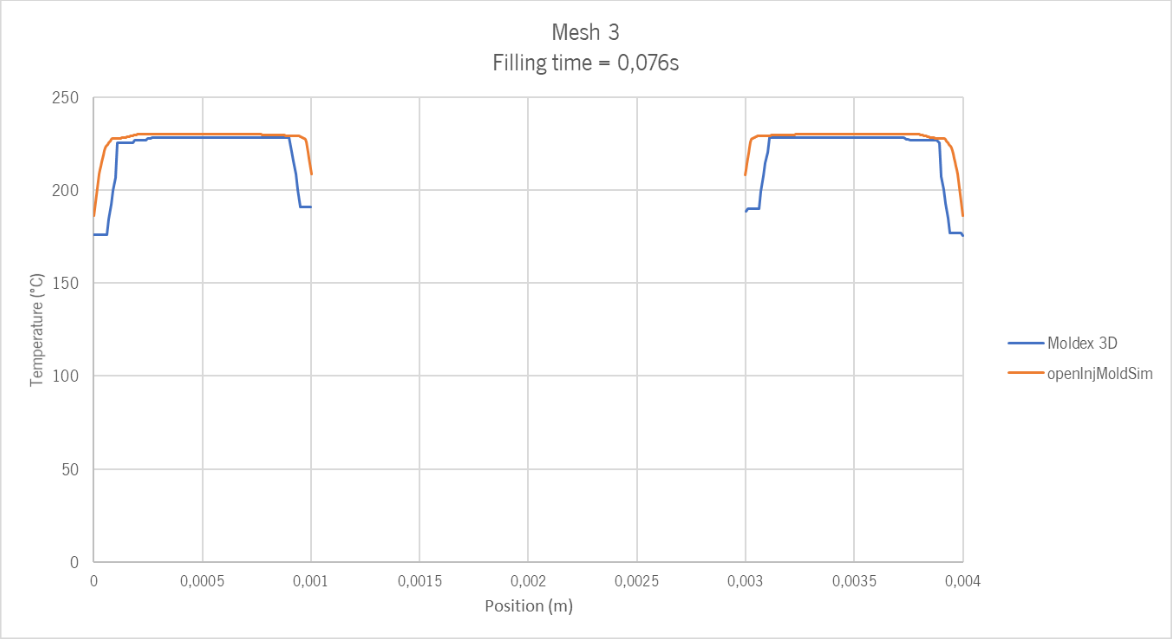

Figure 35 – Temperature predictions by openInjMoldSim and Moldex3D® program at 0,076s of filling time for the Mesh 3.

Also, at 0.076s of filling time, the temperature was analysed in the cross-sections indicated in Figure 36.

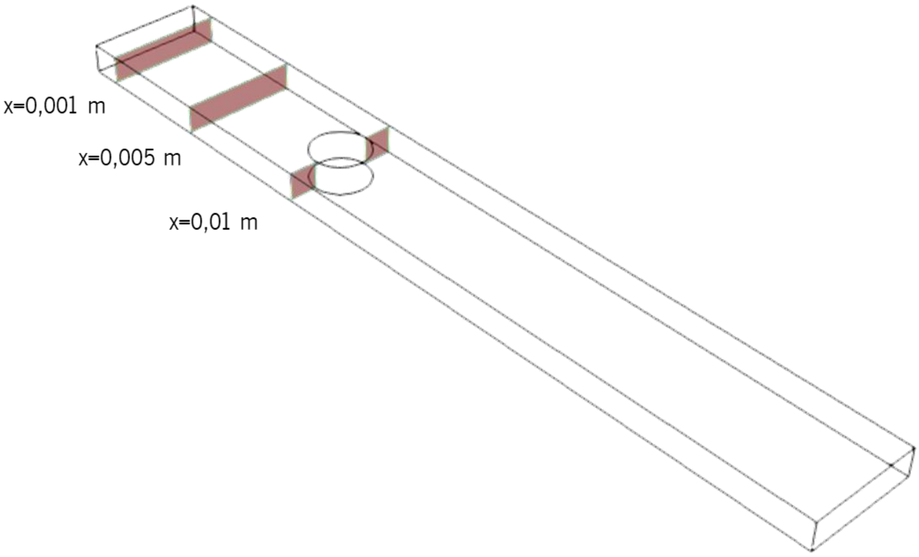

Figure 36 – Cross-section locations for which the results were analysed.

In Figure 37 the temperature results at cross-sections for Moldex3D® and openInjMoldSim are presented, for Mesh 1 at 0,076s of filling time. The results obtained in both programs are in general similar. Contrary to what the graphics analysed previously demonstrate, in case of Moldex3D® the temperature next to the walls does not remain constant, showing a variation of the temperature. In case of openInjMoldSim the same variation is verified.

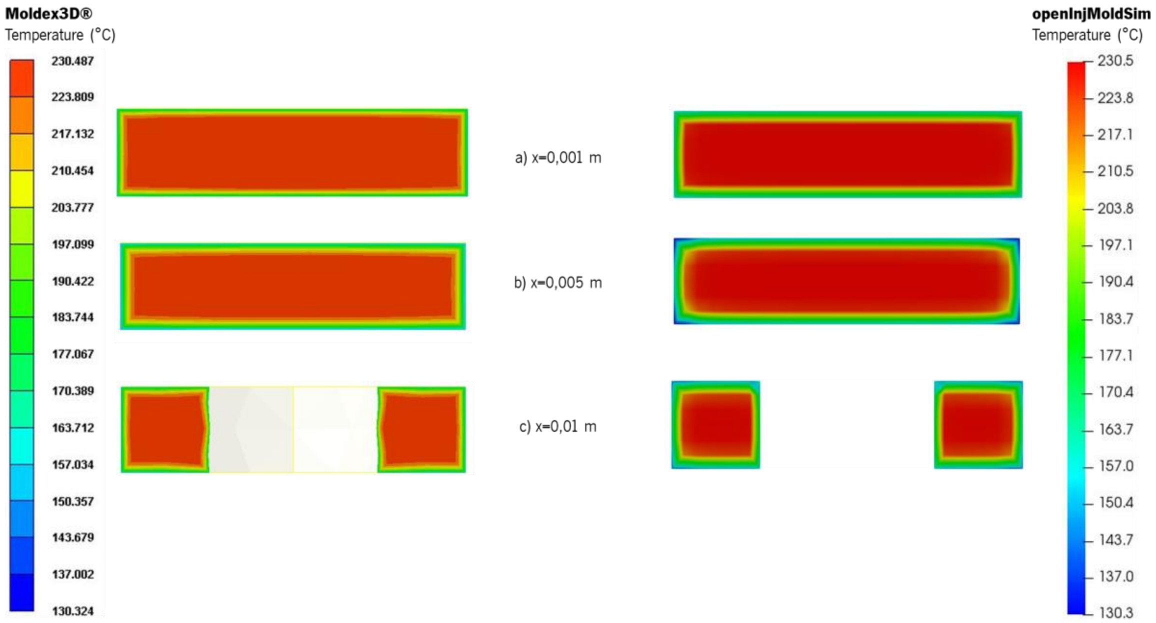

Figure 37 –Temperature distribution for Mesh 1 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.

In Figure 38 the temperature results at cross-sections for Moldex3D® and openInjMoldSim are presented, for Mesh 2 at 0,076s of filling time. The results obtained in both programs are in general similar and with the refinement of the mesh temperature variations became smoother. In the same way that had occurred for the Mesh 1 in the case of Moldex3D®, the temperature profile of the cross-section shows a variation of the temperature next to the walls, in contrary to the results given in Figure 34.

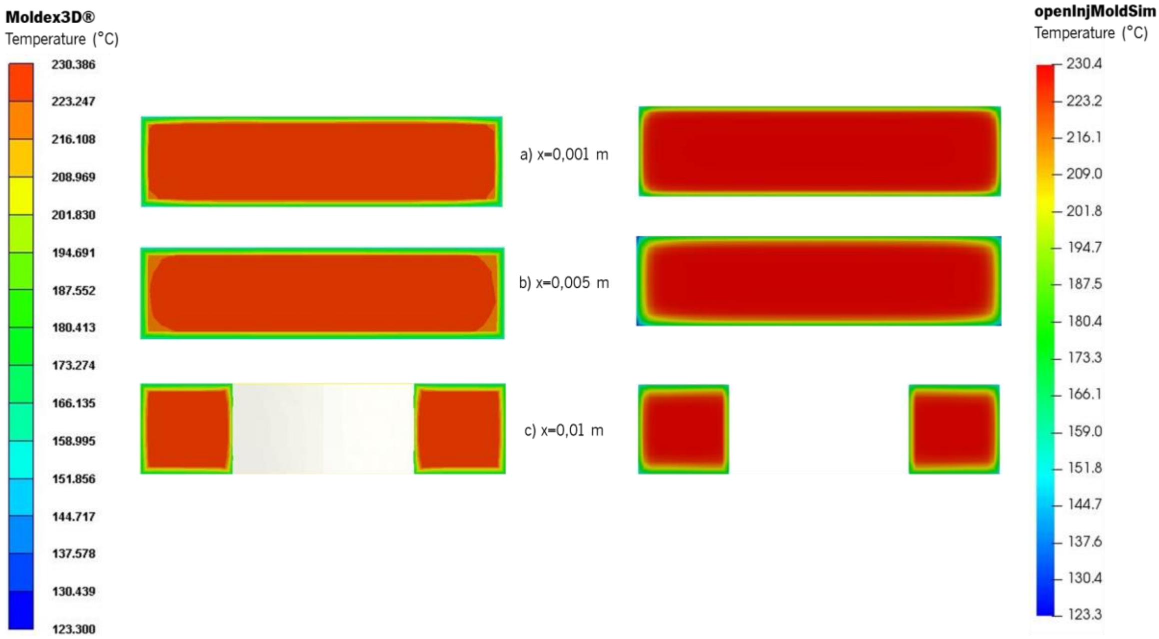

Figure 38 –Temperature distribution for Mesh 2 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.

Once again with the refinement of the mesh, the temperature variations next to the wall become softer and the overall profile of both programs remains very similar, as shown in Figure 39.

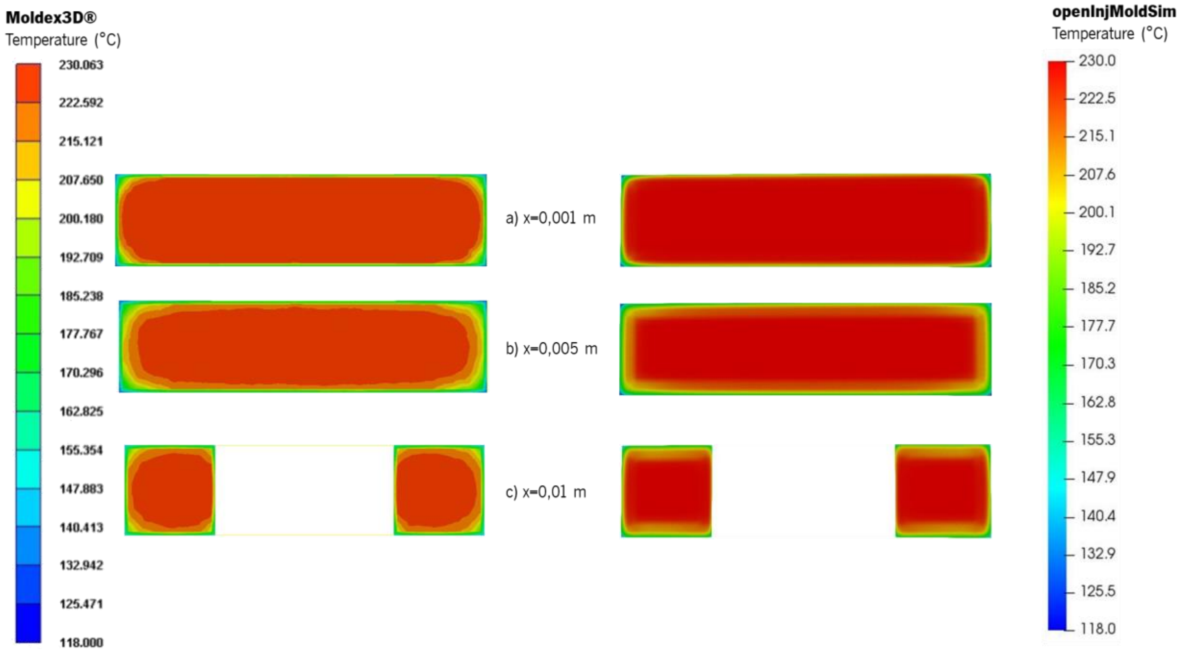

Figure 39 –Temperature distribution for Mesh 3 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.

As the cross-section results appeared not to be the same and show some inconsistency in the Moldex3D® case, the results along the line A – B were superimposed with the ones obtained for the cross-section distribution, at the same positions (x=0,01 m), as shown in Figure 40.

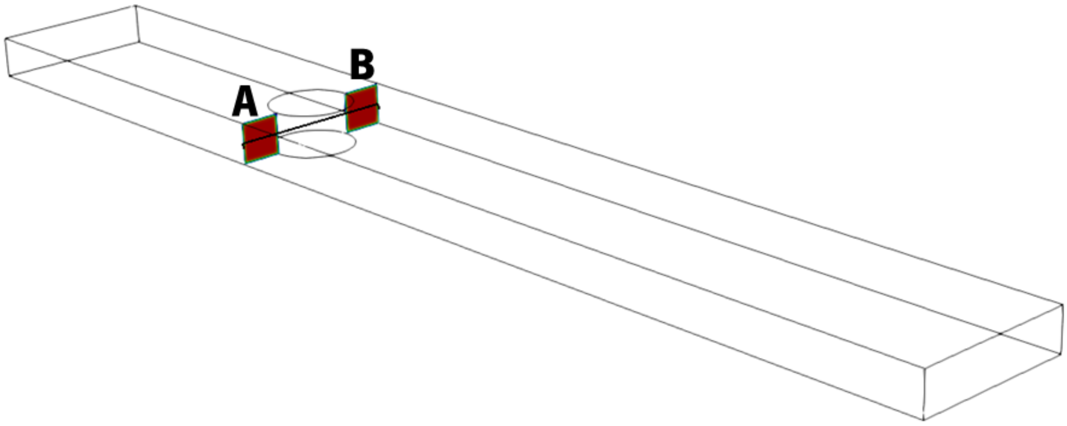

Figure 40 – Position of cross-section and line where the results were compared for Moldex3D®.

In Figure 41 the results of the Moldex3D® for Mesh 1 are presented and it can be observed that, in fact, there is no coherence between results. In Figure 42 the results of openInjMoldSim for Mesh 1 are presented and it can be observed that both results are in accordance, the temperature variation is the same in both representations.

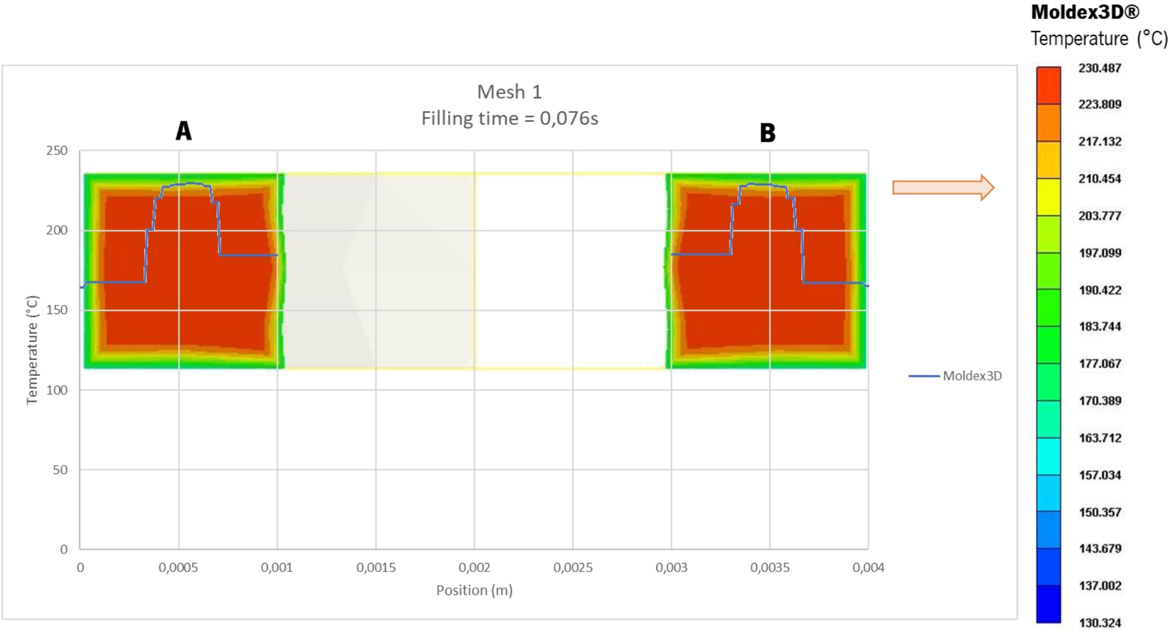

Figure 41 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 1.

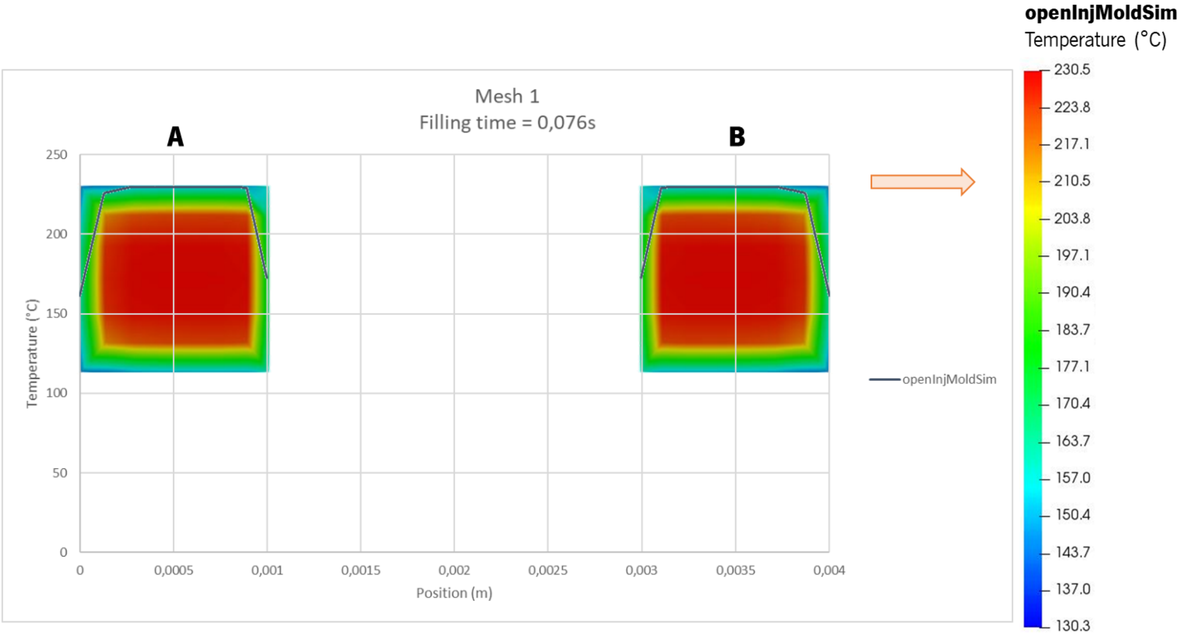

Figure 42 –Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 1.

Figure 43 presents the comparison between cross-section temperature distribution at x=0,01m and results along the line A – B, for Moldex3D® for Mesh 2. Despite the refinement of the mesh the graphs along the line continue to be inconsistent with the ones obtained for the cross-section temperature distribution. The results presented in Figure 44 for openInjMoldSim for Mesh 2 where, once again, it is shown that there is consistency for both presentations.

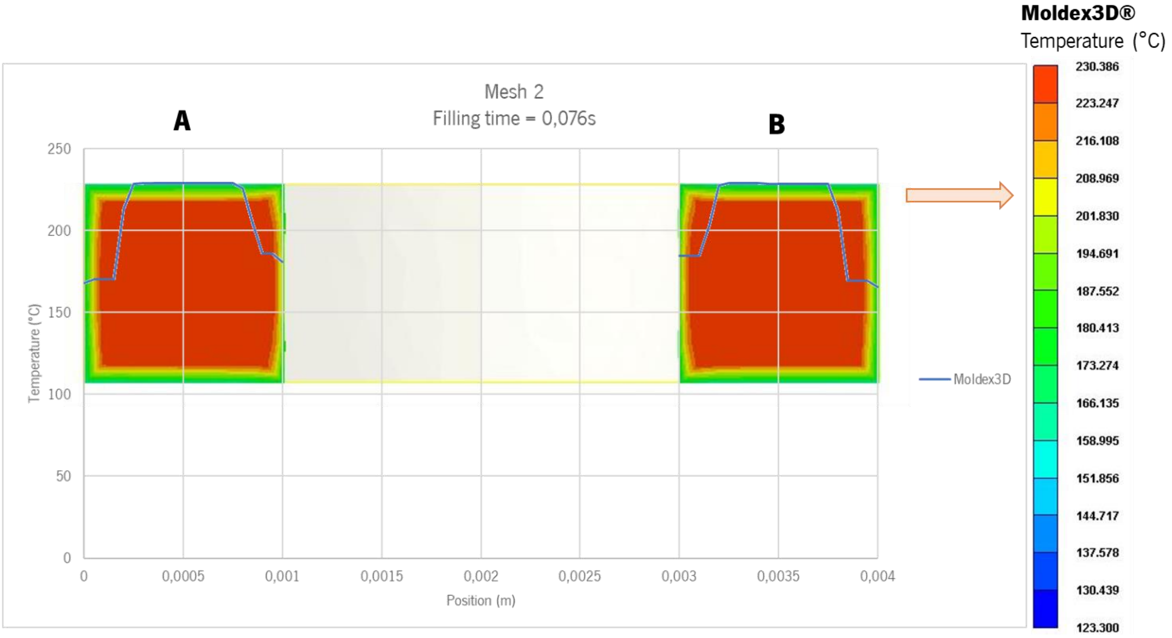

Figure 43 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 2.

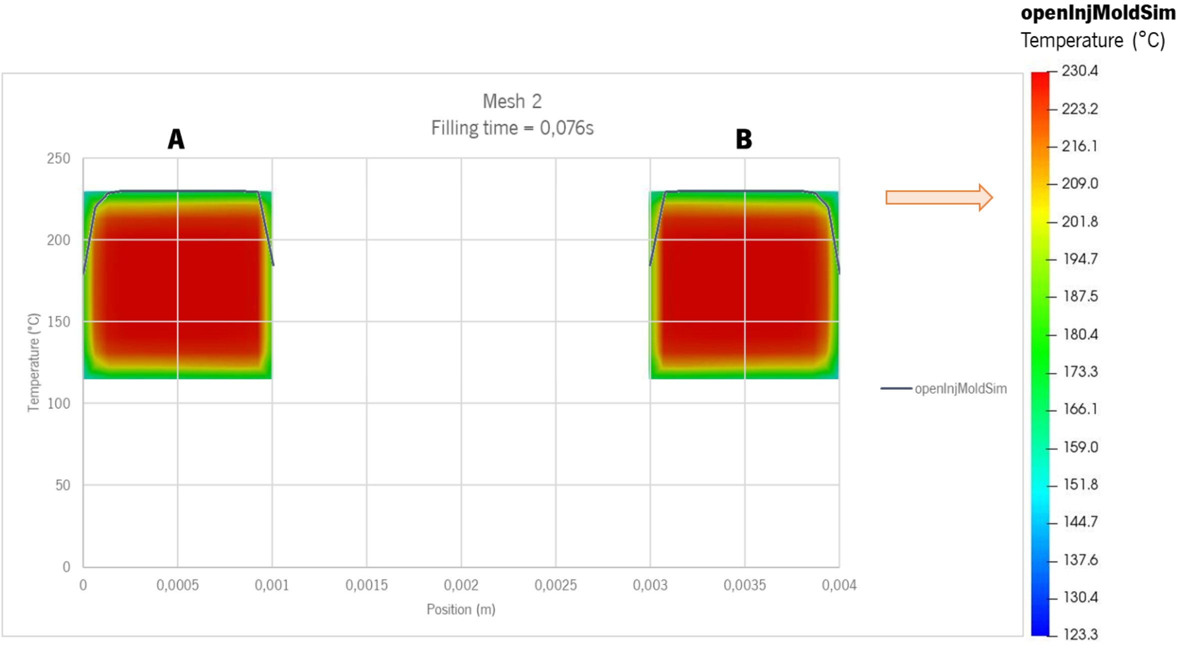

Figure 44 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 2.

In the same way as in Mesh 2, the results obtained for Mesh 3 are incoherent in Moldex3D® (see Figure 45), but in openInjMoldSim the coherence remains (see Figure 46). There is no explanation for this incoherence between results in Moldex3D®.

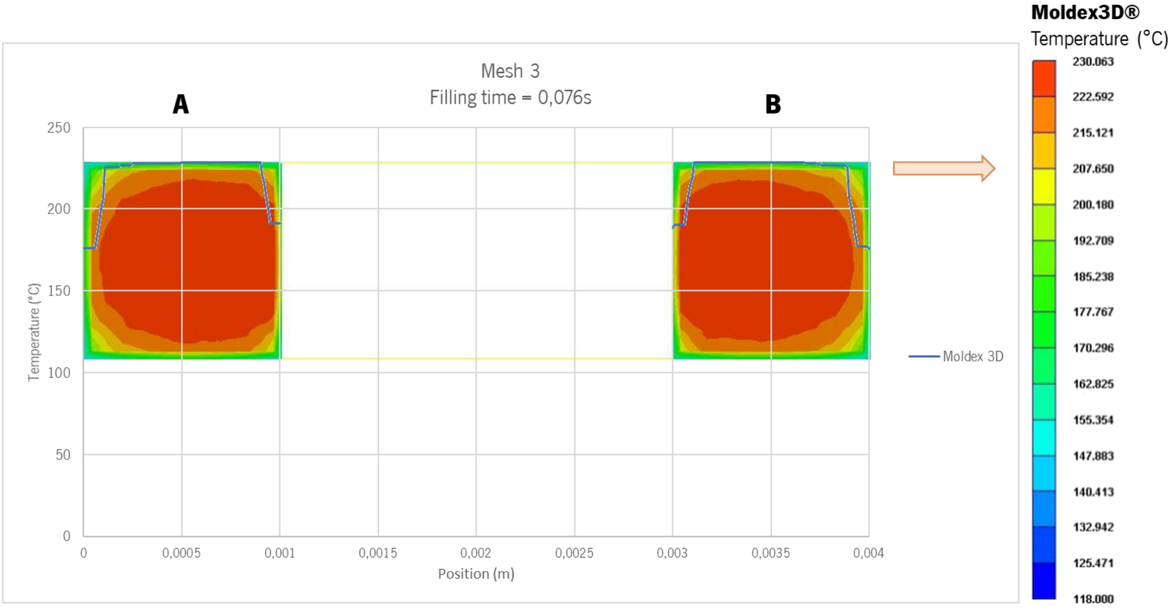

Figure 45 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 3.

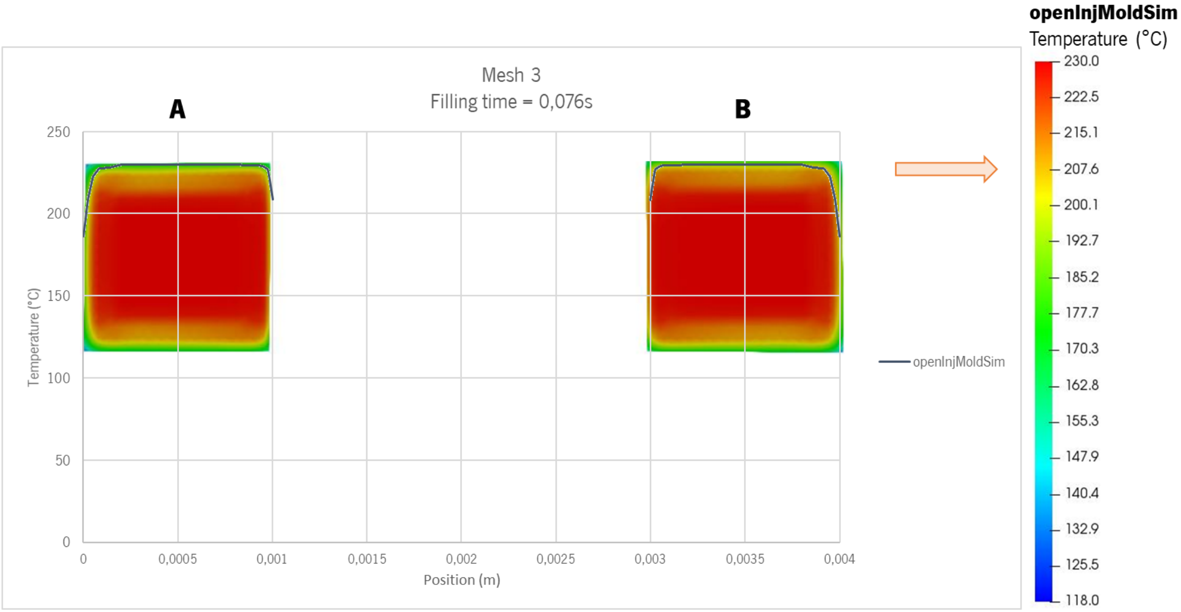

Figure 46 – Comparison between cross-section temperature distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 3.

Due to the incoherence of the results in Moldex3D®, for the comparison of results between the two programs were only used the temperature distribution in the cross-section. This is because as the results along the line, for Moldex3D®, were drawn by several points along the same and it is not known how the program calculates the value of the point may justify why these results did not represent what actually happens during the filling phase. Thus, the results that were considered to compare the two programs were the results obtained by the cross-section. Comparing them for the same refinement level, it is possible to observe that they are quite similar (see Figures 37, 38 and 39). The temperature next to the external walls has a similar variation and close values between the two programs. With the refinement of the mesh the variations of temperature next to the walls become smoother, as expected.

3.3 ANALYSIS OF VELOCITY RESULTS

The analysis performed for the temperature distribution, presented in the previous section, was replicated for the velocity distribution along the line shown in Figure 30, at 0,076s of filling time.

The results obtained with Mesh 1 (see Figure 47) in openInjMoldSim are symmetric between side A and side B. The velocity profile is close to parabolic, on the walls, as it was imposed, velocity is zero. In the centre of side A and B the maximum velocity is reached with an approximate value of 0,80 m/s. In the program Moldex3D®, it was expected that the behaviour of the velocity was the same, but this is not verified. Next to the walls the velocity profile presents a plateau in which the velocity is null, so it is expected, based on the no-flow velocity, that the material will already have solidified. However, if this was the case, since the flow rate imposed is the same in both programs it is expected that the maximum velocity in Moldex3D® was much higher than the maximum velocity given by openInjMoldSim, because the area through which the flow passes in the Moldex3D® would be minor than the area that the flow passes in the openInjMoldSim, but this did not occur. The maximum value of velocity in Moldex3D® is approximately 0,55 m/s on side A and 0,48 m/s on side B, lower values than the maximum velocity reached by the program openInjMoldSim (u=0,811 m/s). As happened for the temperature distribution, the mesh refinement had a huge effect on the velocity distribution predicted by Moldex3D®.

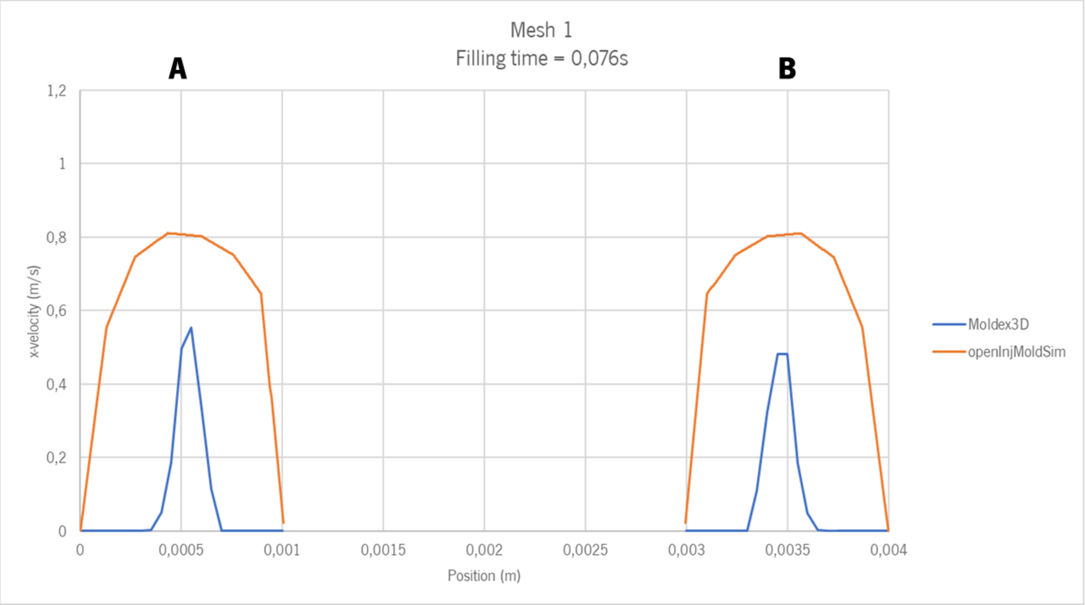

Figure 47 – Flow direction velocity component distribution given by openInjMoldSim and Moldex3D®, at 0,076s of filling time for the Mesh 1.

The results in Mesh 2 (Figure 48), in openInjMoldSim the results are similar to those in Mesh 1. The difference is in the maximum velocity reached which is about 0,87 m/s. This increase indicates that the meshes are not yet converged. In the program Moldex3D®, the results continue to present the null-velocity plateau next to the walls, but this time smaller ones. Thus, the area through which the flow passes is smaller in Moldex3D® than in openInjMoldSim, so as expected the maximum velocity reached in Moldex3D® is higher, being approximately 1m/s on both sides. The velocity profile, in the case of the commercial program, remain plug like, and physically a parabolic shaped profile was expected.

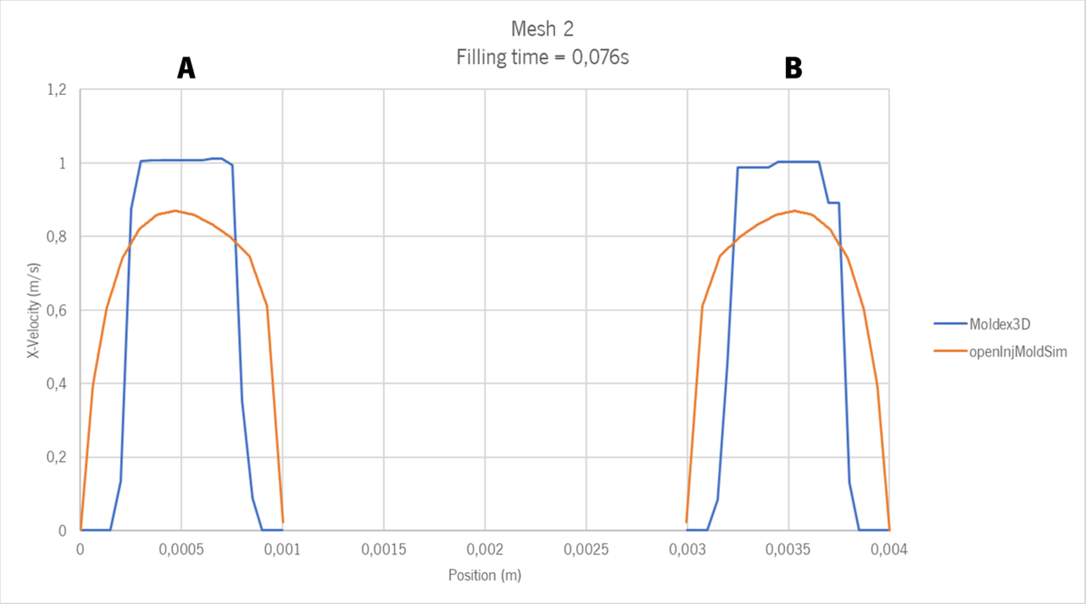

Figure 48 – Flow direction velocity component distribution given by openInjMoldSim and Moldex3D®, at 0,076s of filling time for the Mesh 2.

In Figure 49 the results of openInjMoldSim for the Mesh 3 are presented, which are similar to those of the other meshes. The difference is in the maximum velocity reached which is about 0,95 m/s. This increase indicates that the meshes are not yet converged. In the program Moldex3D®, the results continue to present the zero-velocity plateau next to the walls, but, as expected, smaller than the others presented in the previous results, the maximum velocity reached was equal to that obtained by the results of the Mesh 2 u=1m/s. The velocity profile remains non-parabolic, showing a strange stepwise distribution.

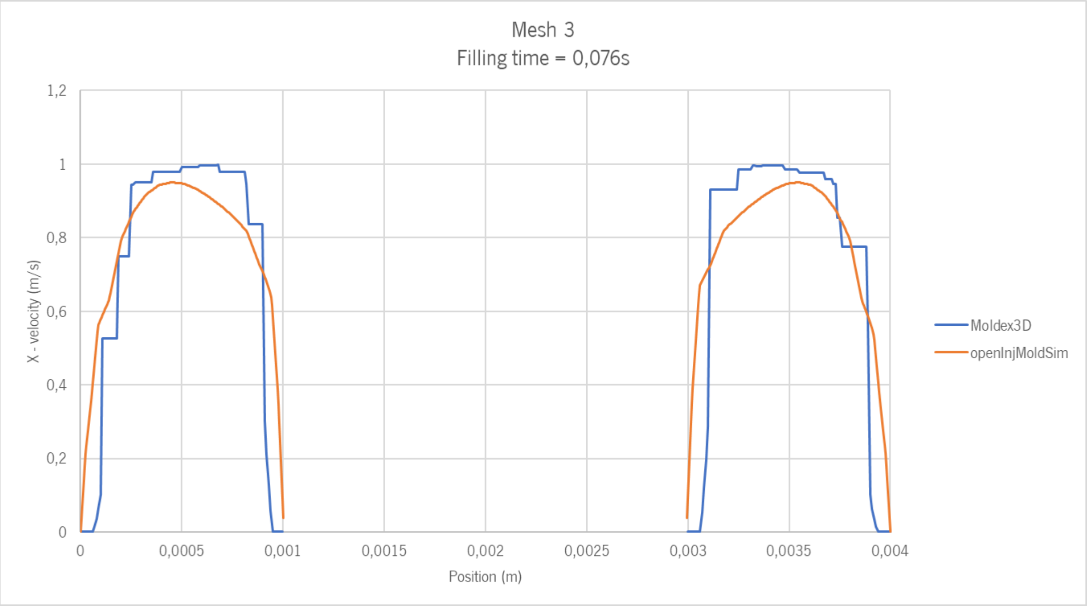

Figure 49 – Flow direction velocity component distribution given by openInjMoldSim and Moldex3D®, at 0,076s of filling time for the Mesh 3.

In Figure 50, a comparison is made between the temperature and velocity distribution in the cross-section x=0,01m, for Mesh 3 given by Moldex3D®. This comparison is made to verify if the null velocity next to the walls is justified by the temperature contour plots. Considering that the temperature at which the material is next to the outer walls is T=178°C and the inner wall next to the cylinder is T=197°C, which are both higher than the temperature at which the material solidifies (T_{freeze} =101°C), these null velocity results near the walls are unphysical.

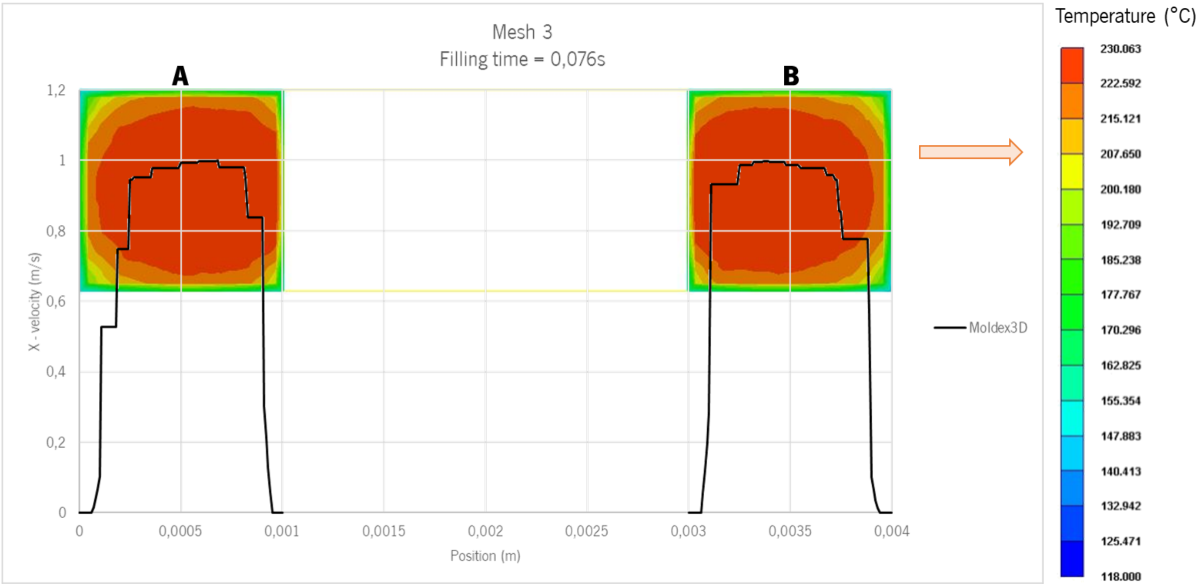

Figure 50 – Comparison between temperature distribution in cross-section x=0,01m and x-velocity results for Moldex3D at 0,076s of filling time for the Mesh 3.

As in the case of temperature, axial velocity component distribution was also analysed at the cross-sections shown in Figure 36. For Mesh 1, given in Figure 51, the velocity profile for openInjMoldSim appears to be parabolic in contrast to what happens in the case of Moldex3D®. At the filling time 0,076s, in the case of Moldex3D® the maximum x-velocity changes from u=0,46m/s to u=1,022 m/s from the position x=0,001m to the position x=0,01m, respectively. In the case of openInjMoldSim the maximum velocity changes from u=0,392 m/s to u=0,811 m/s. These velocity increases due to the cross-section reduction. The velocity is higher in Moldex3D® because, next to the walls, the velocity is null, that is, the material seems to be solidified and therefore the actual flow cross-section is smaller. As shown in these results the ones presented in Figure 47, previously analysed, seem incorrect. Moreover, the Moldex3D® contour plots seem to correspond to a significantly coarser mesh, when compared to the ones provided by openInjMoldSim, which might be an indication that the mesh details provided by the former are incorrect.

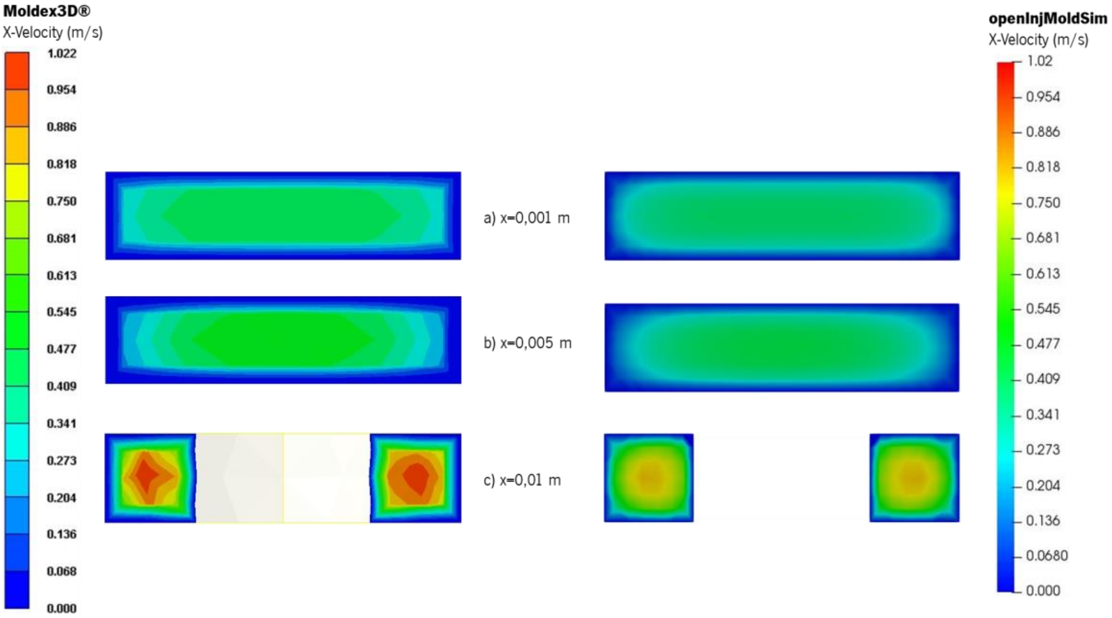

Figure 51 – Velocity contours for Mesh 1 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.

The Mesh 2 and 3 results are presented in Figure 52 and Figure 53. As in the case of Mesh 1, the maximum velocity occurs in the centre of the cross-section at x=0,001m (see Figure 52 c) and Figure 53 c)). The maximum velocity reached in the case of the Moldex3D® was u=1,027 m/s and in the case of openInjMoldSim was u=0,869 m/s, for Mesh 2. For Mesh 3 in the case of the Moldex3D® was u=0,99 m/s and in the case of openInjMoldSim was u=0,95 m/s. With the refinement of the mesh, in Moldex3D® the velocity distribution has become smoother, and the profile begins to become more parabolic. As in Mesh 1, the results from Moldex3D® did not appear to be consistent with the results along the line, the same happens for the Mesh 2 and 3 and therefore the results compared hereafter. Also, once again the Moldex3D® contour plots seem to correspond to a significantly coarser mesh, when compared to the ones provided by openInjMoldSim, which might be an indication that the mesh details provided by the former are incorrect.

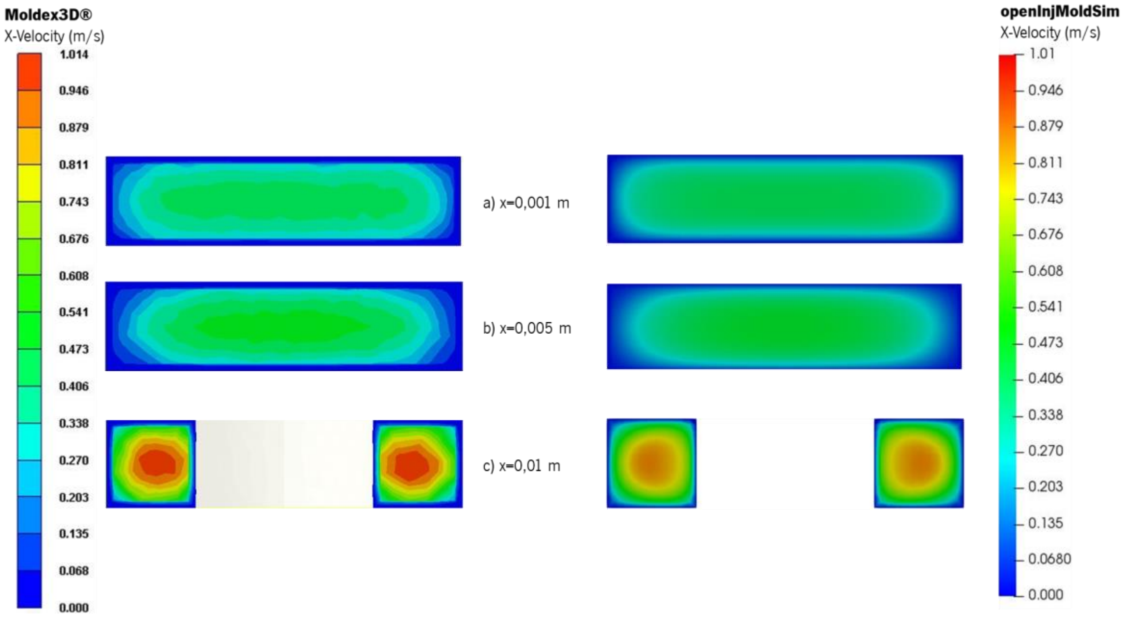

Figure 52 – Velocity contours for Mesh 2 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.

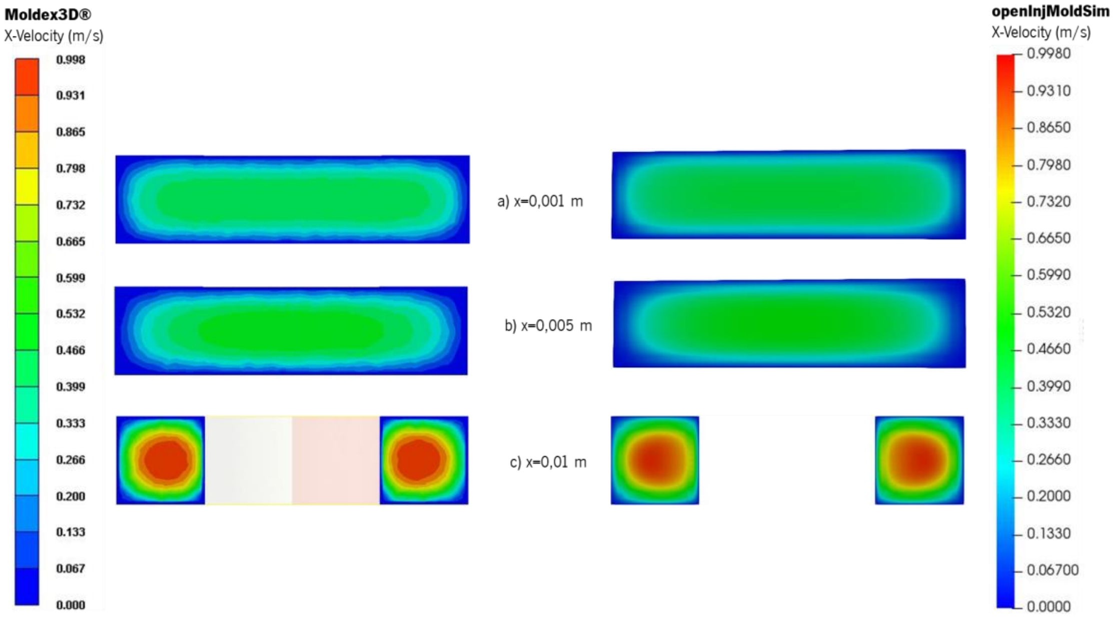

Figure 53 – Velocity contours for Mesh 3 at cross-section a) x=0,001m b) x=0,005m and c) x=0,01m in both programs.

The results of the x-velocity distribution at cross-section x=0.001 m and the results along the line (see Figure 40) for Mesh 1 of Moldex3D® and openInjMoldSim are shown, respectively, in Figure 54 and Figure 55. It is possible to observe that the results are not coherent, the x-velocity, according to the results along the line, is zero at a much greater thickness than that which appears in the cross-section velocity distribution. In the case of openInjMoldSim, through Figure 55 it is possible to observe that there is coherence in the results, the maximum velocity reached is the same for both results u=0,811 m/s. In the case of Moldex3D® along the line results the maximum x-velocity is u=0,55 m/s and u=0,58 m/s, side A and B respectively and in cross-section results the maximum x-velocity is u=1,022 m/s.

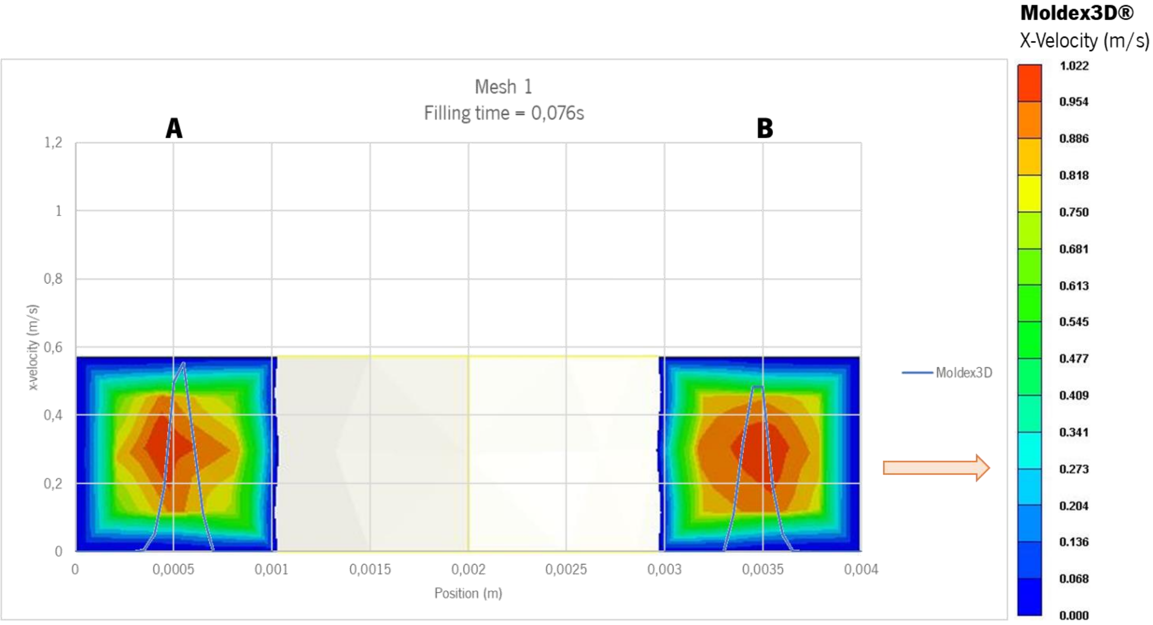

Figure 54 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 1.

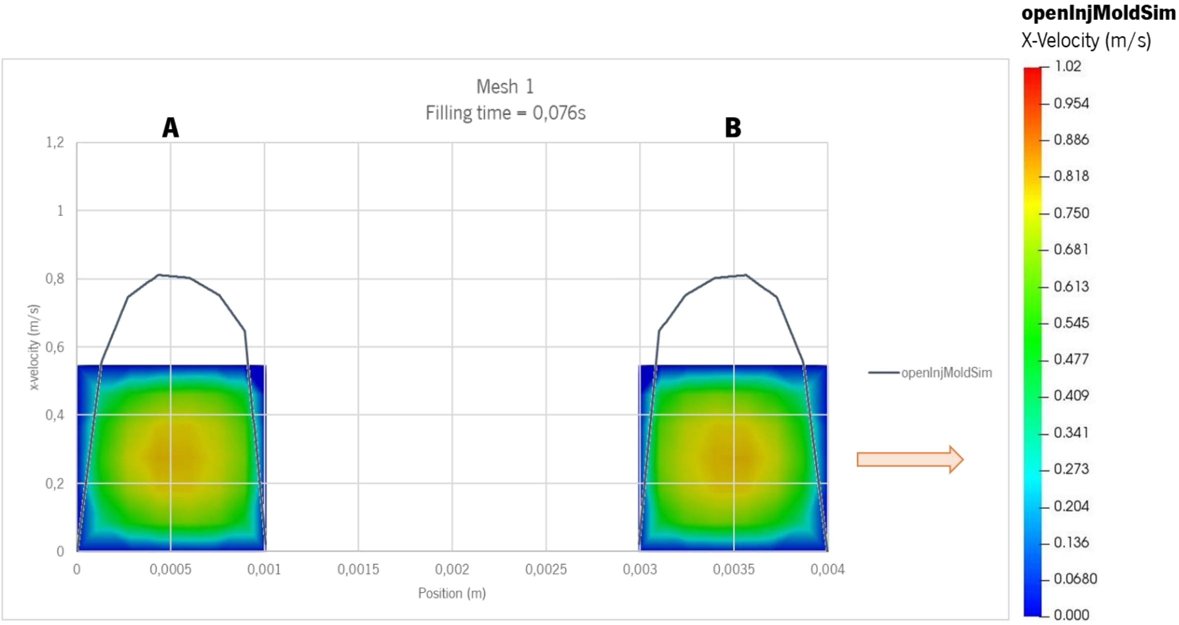

Figure 55 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 1.

The results of the x-velocity distribution at cross-section x=0,01 m and the results along the line (see Figure 40) for Mesh 2 of Moldex3D® and openInjMoldSim are shown, respectively, in Figure 56 and

Figure 57. Once again in the case of Moldex3D® there is no coherence between the results of the cross-section and along the line. The maximum velocity reached in the results of the cross-section is u=1,027m/s and along the line is u=1,01 m/s. In case of openInjMoldSim the maximum velocity is u=0,869 m/s in both results.

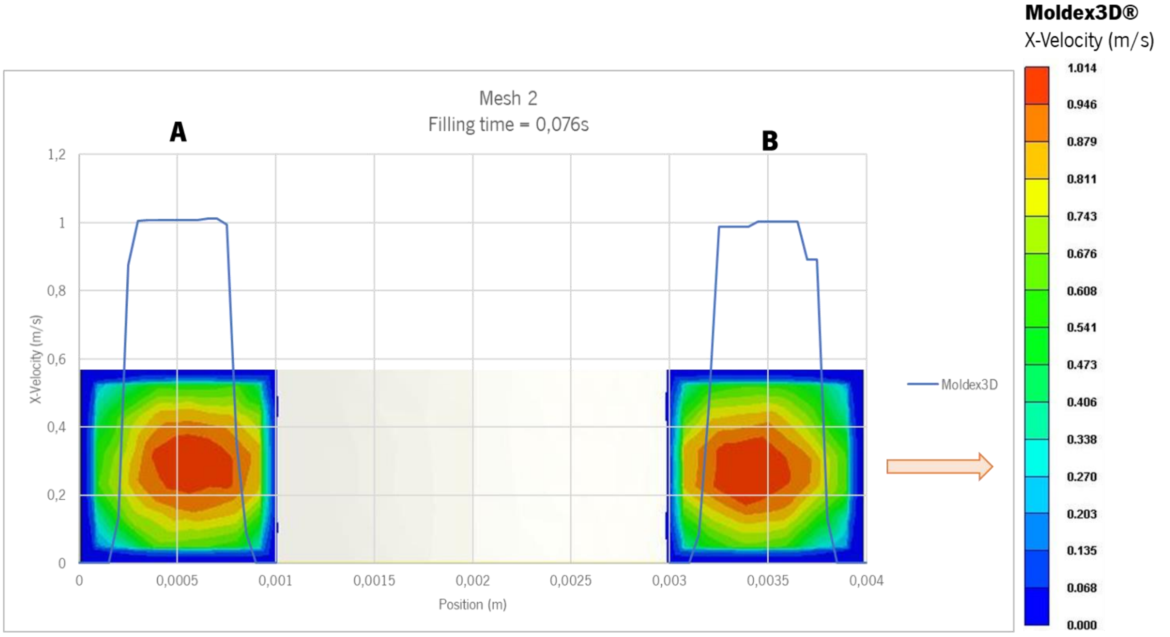

Figure 56 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 2.

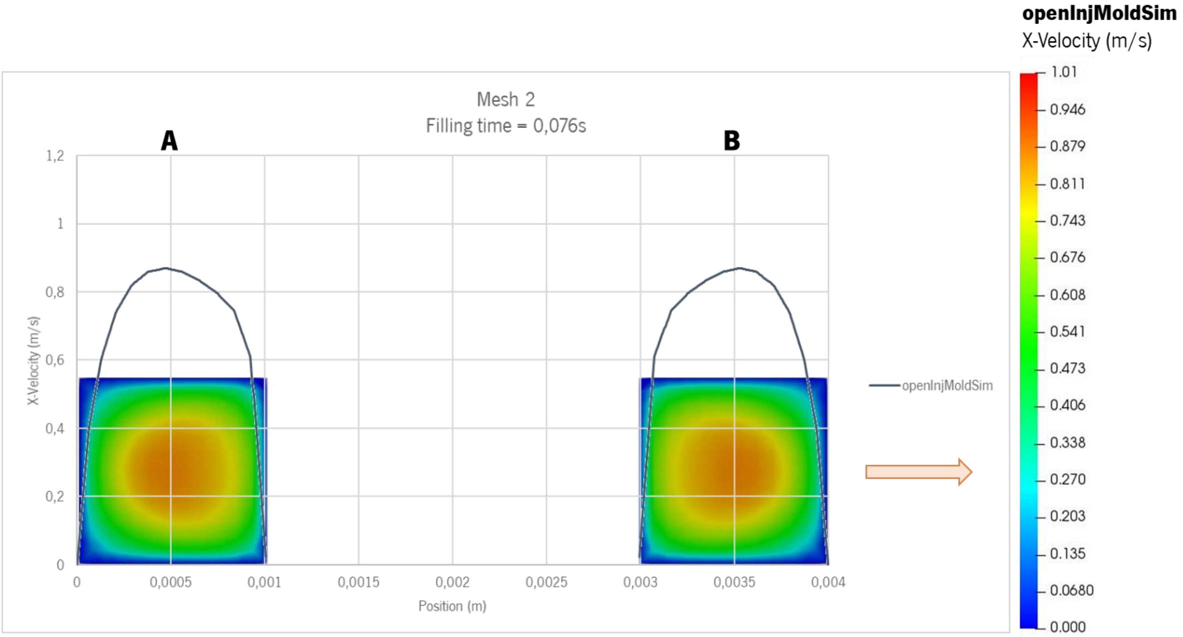

Figure 57 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 2.

In Figures Figure 58 and Figure 59 the results of the x-velocity distribution at cross-section x=0.001 m and the results along the line (see Figure 40) for Mesh 3 of Moldex3D® and openInjMoldSim are respectively shown. Once again in the case of Moldex3D® there is no coherence between the results because the profile of the results along the line has several breaks and it is not parabolic as shown in the cross-section velocity distribution.

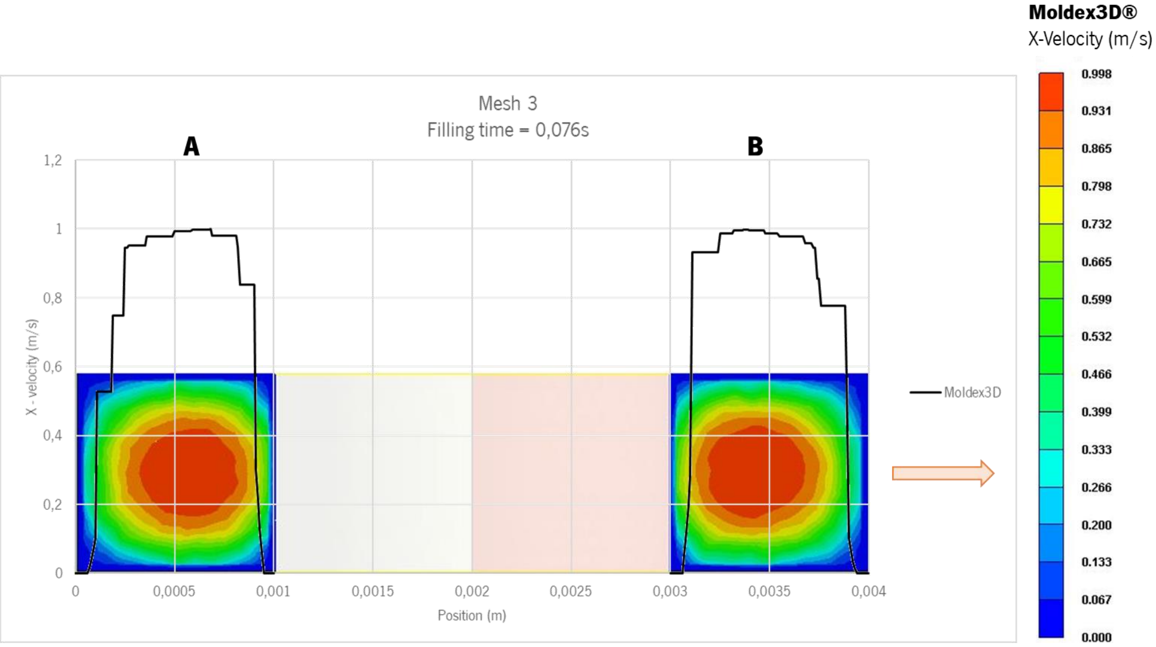

Figure 58 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for Moldex3D® for Mesh 3.

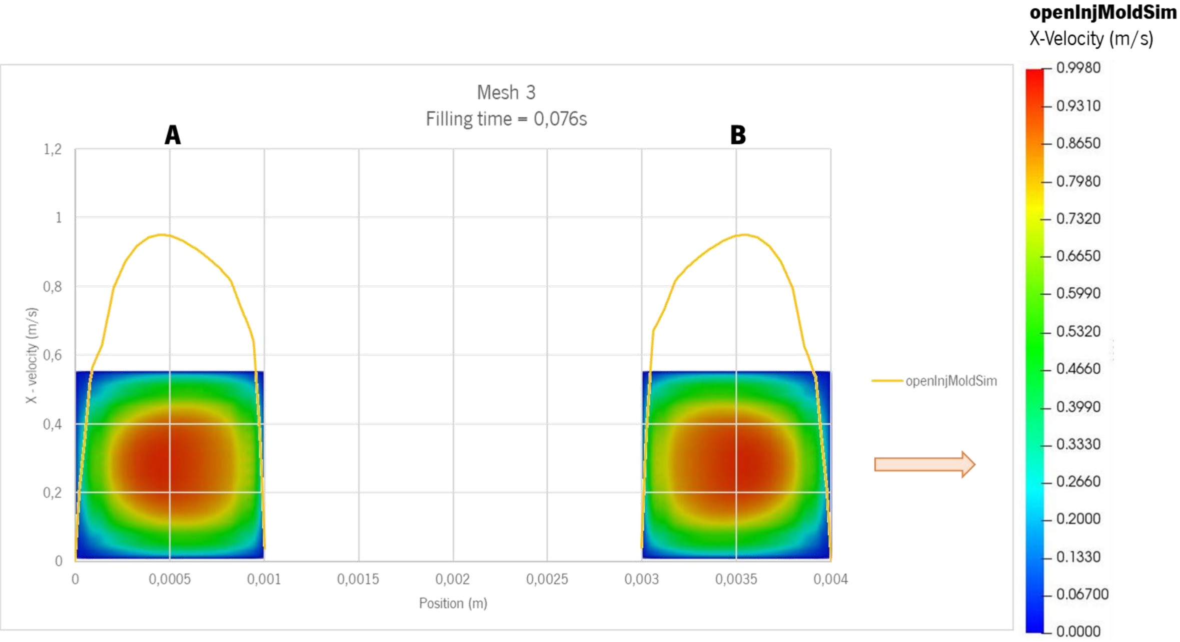

Figure 59 – Comparison between cross-section x-velocity distribution at x=0,01m and results along the line where the results were studied, for openInjMoldSim for Mesh 3.

The results that best represent what is happening in practice, in the case of Moldex3D®, are the cross-section results, so these are the ones that were compared between the two programs. Table 11 shows the values of maximum x-velocities reached for the two meshes in both programs, using as reference the average value. With the refinement of the mesh the relative difference decreases, however the mesh convergence seems have not been achieved yet.

Table 11 – Maximum x-velocities reached for the two meshes in both programs with the relative difference based on the average values.

	Moldex3D	**openInjMoldSim**	**Relative difference**
Mesh 1	$u=1{,}022\ \mathrm{m/s}$	$u=0{,}811\ \mathrm{m/s}$	$\frac{1{,}022-0{,}811}{0{,}9165}\times 100=23\%$
Mesh 2	$u=1{,}027\ \mathrm{m/s}$	$u=0{,}869\ \mathrm{m/s}$	$\frac{1{,}027-0{,}869}{0{,}948}\times 100=17\%$
Mesh 3	$u=0{,}99\ \mathrm{m/s}$	$u=0{,}95\ \mathrm{m/s}$	$\frac{0{,}99-0{,}95}{0{,}97}\times 100=4\%$

3.4 ANALYSIS OF PRESSURE RESULTS

In Figure 60 are presented the pressure contours at the end of filling time (0,1s), are presented, for both programs for Mesh 1. The pressure distribution along the flow axis is predicted to be approximately linear by both programs, but the highest value of pressure is different. The pressure drop in Moldex3D® is Δp=5,498 MPa and in openInjMoldSim Δp=8,43 MPa, as this value is very high, compared to the Moldex3D®, the pressure drop was analysed for previous times as shown in the graph plotted in Figure 61. From the graph it is possible to see that in the previous instants the pressure drop is less than half (Δp≈4 MPa) which shows that at 0,1s of filling time corresponds to the terminal phase of filling, although the material is compressible it is poorly compressible, causing a sudden increase of the pressure.

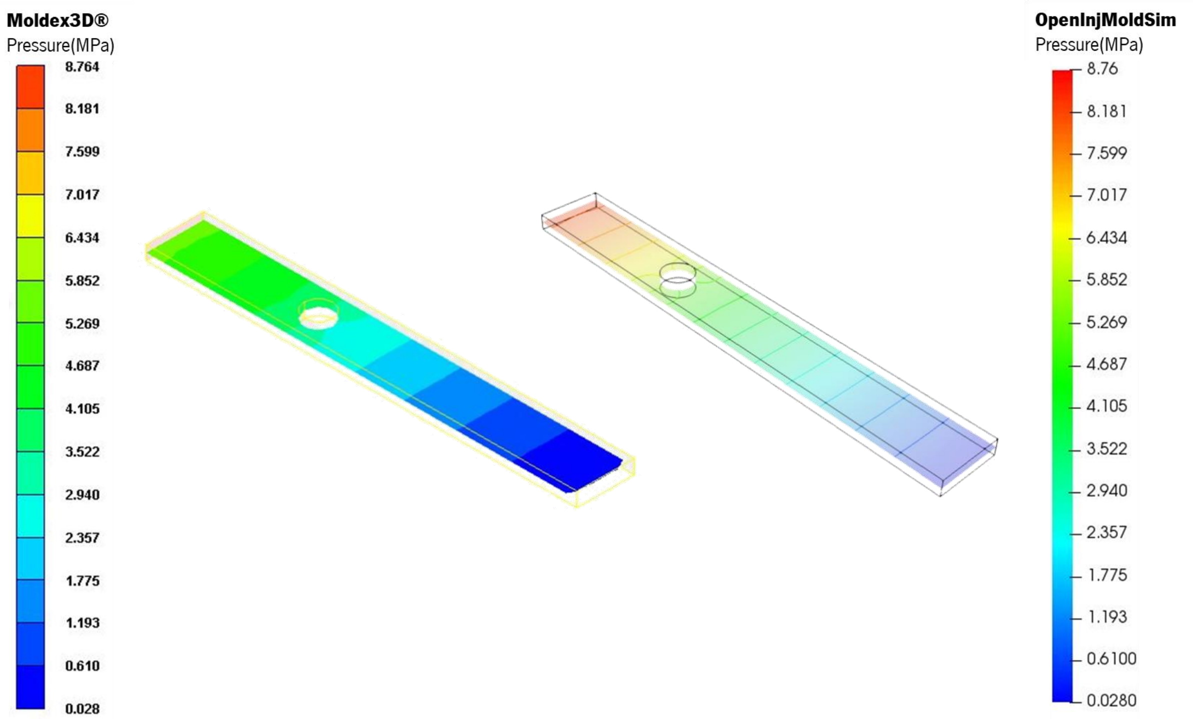

Figure 60 – Pressure contours for Mesh 1 at the end of filling time.

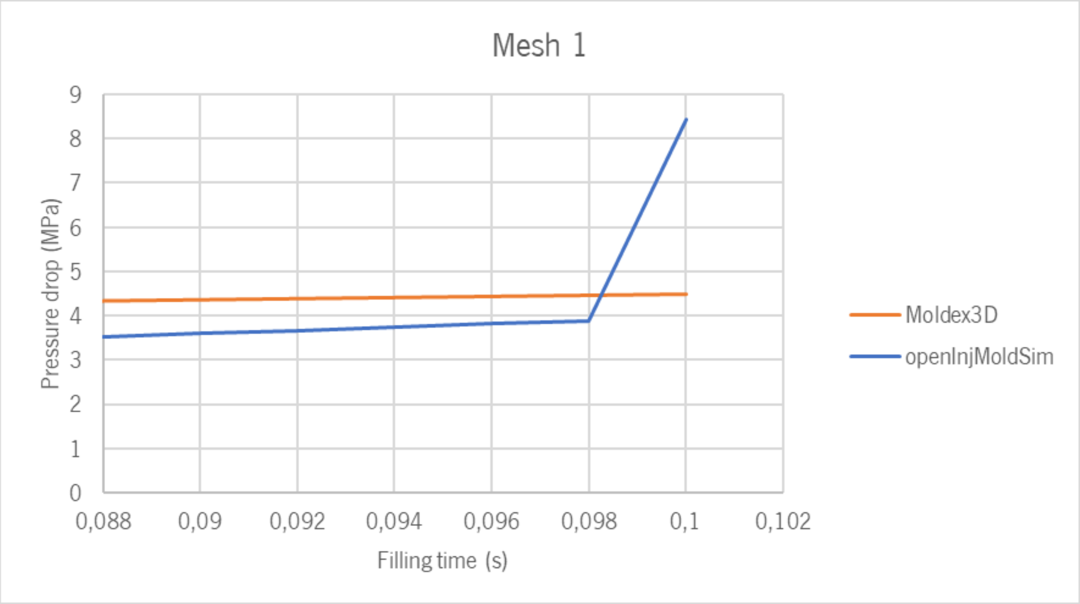

Figure 61 – Pressure drop variation for Mesh 1 in both programs.

For Mesh 2, the pressure contours at the end of filling time are shown in Figure 62, again the pressure distribution along the flow is linear in both programs the highest value of pressure in the two programs is that it differs. The pressure drop in Moldex3D® is Δp=5,907 MPa and in openInjMoldSim Δp=4,39 MPa. In the graph present in Figure 63 it is possible to observe that the pressure drop increases linearly for both programs, in contrast to what happened in the Mesh 1 that there was an exponential increase of the pressure drop in openInjMoldSim results.

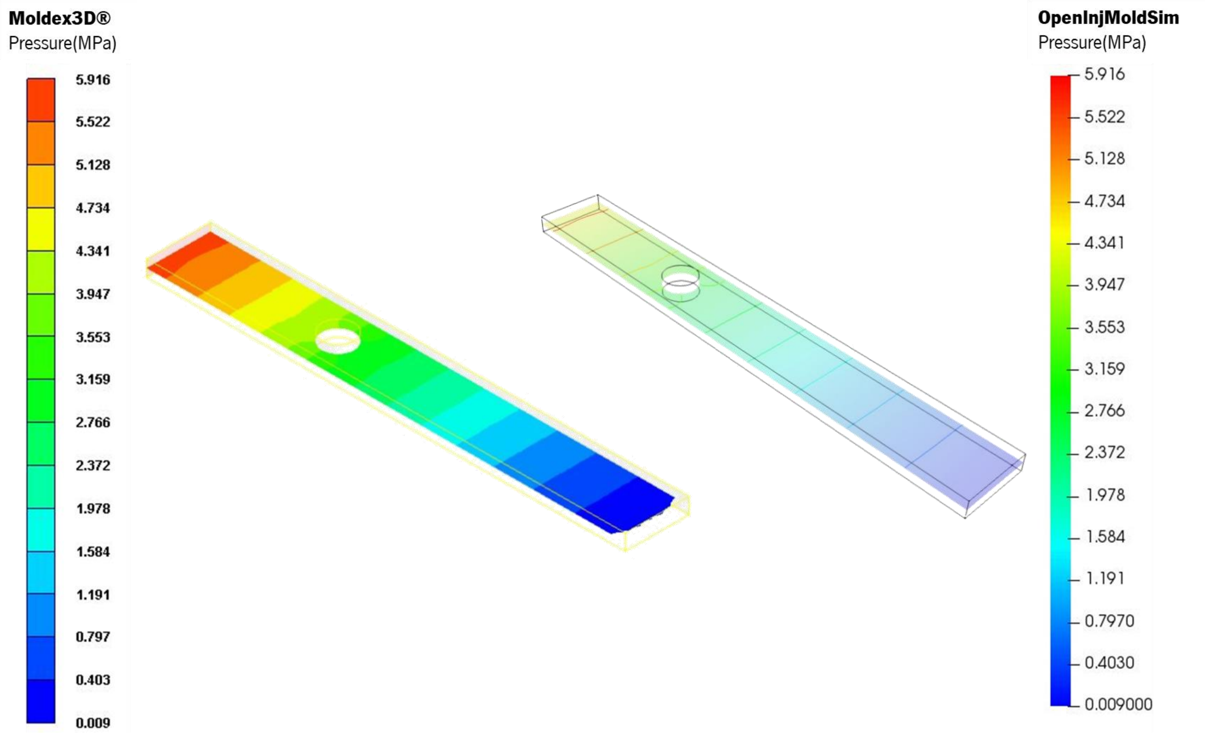

Figure 62 – Pressure contours for Mesh 2 at the end of filling time.

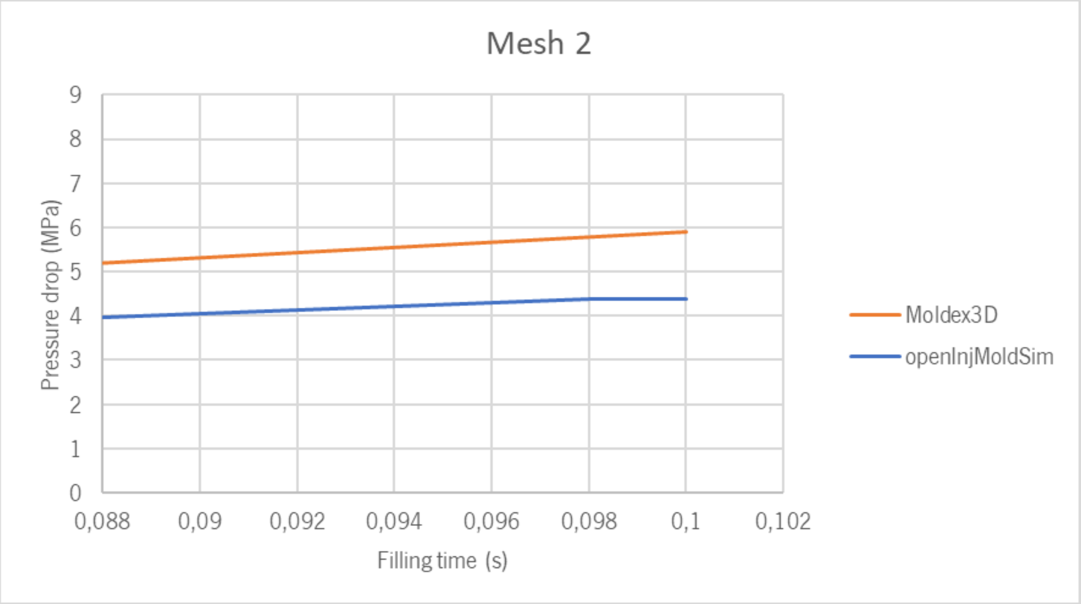

Figure 63 – Pressure drop variation for Mesh 2 in both programs.

For Mesh 3, the pressure contours at the end of filling time are shown in Figure 64, once again the pressure drop is more or less linear in both programs the highest value of pressure in the two programs is that it differs. The pressure drop in Moldex3D® is Δp=6,204 MPa and in openInjMoldSim Δp=5,288 MPa. Once again, the pressure drop increases linearly, as can be observed in Figure 65, for both programs.

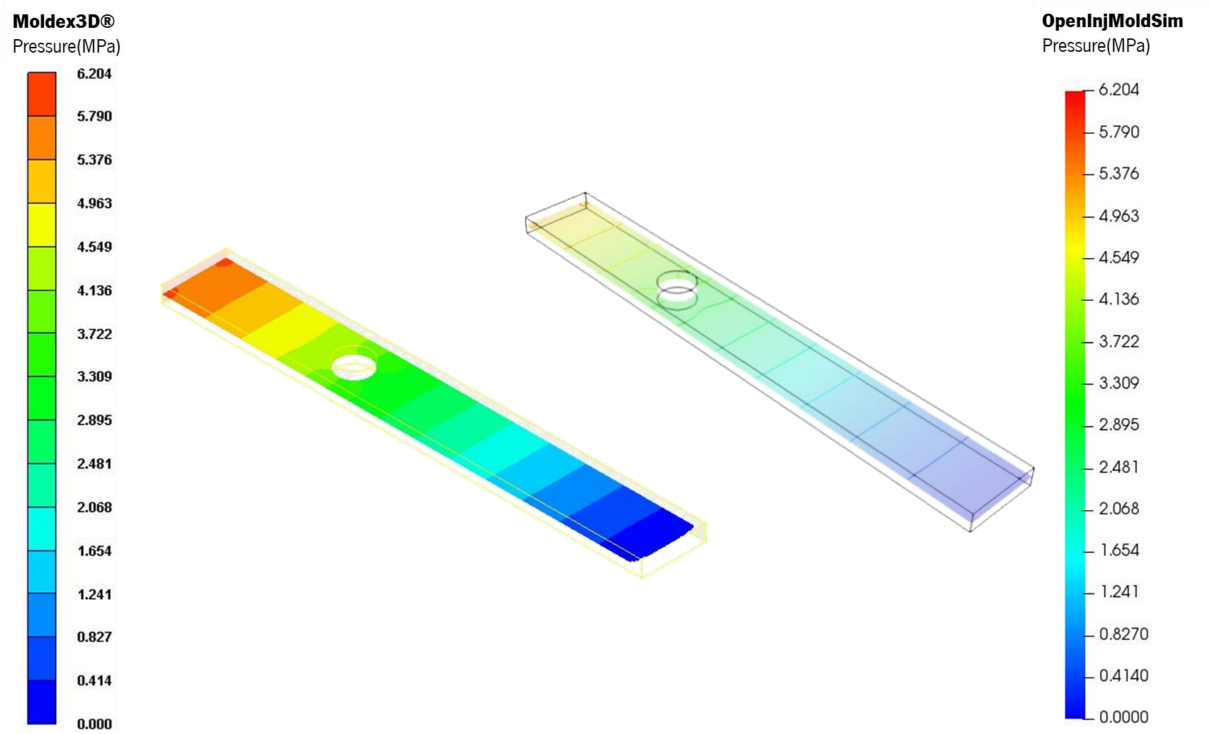

Figure 64 – Pressure contours of Mesh 3 at the end of filling time.

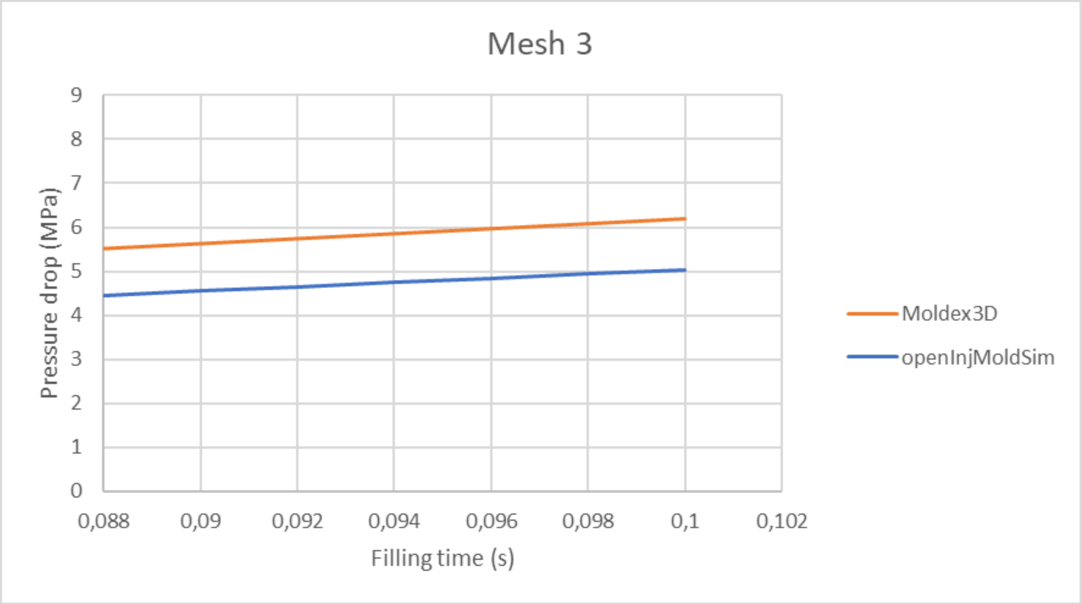

Figure 65 – Pressure drop variation for Mesh 3 in both programs.

Table 12 shows the values of pressure drop for the three meshes in both programs, using as reference the average values. With the refinement of the mesh the relative difference decreases. The meshes are not yet converged so the refinement of the mesh could approach the results between the two programs and reduce the relative difference

Table 12 -Pressure drop for meshes 1, 2 and 3 in both programs with the relative difference based on the average values.

	Moldex3D	**openInjMoldSim**	**Relative difference**
Mesh 1	Δp=5,498 MPa	Δp=8,43 MPa	$\frac{8,43-5,498}{6,964}\times 100 = 42\%$
Mesh 2	Δp=5,907 MPa	Δp=4,39 MPa	$\frac{5,907-4,39}{5,148}\times 100 = 29\%$
Mesh 3	Δp=6,204 MPa	Δp=5,288 MPa	$\frac{6,204-5,288}{5,746}\times 100 = 16\%$

CHAPTER V – CONCLUSIONS AND FUTURE WORK

1. CONCLUSIONS

The injection moulding process is an intricate, dynamic, and transient process, involving convoluted melting-flow-pressure-solidification phases, for which appropriate numerical tools can be a useful design tasks support. However, one of the major limitation associated with the use of commercial software is the usually high cost associated with licences, which can be overcome by resorting to open-source/free.

Open-source numerical tools still have many limitations, compared to the commercial software available, in addition to being few applied to the injection moulding process, those that exist are not fully validated and therefore also cannot have an application in the industry, thus the main objective of this work was to validate the open-source program openInjMoldSim [99, 103] and, for this, the results obtained through this program were compared with ones obtained with a commercial software widely employed in industry, the Moldex3D® [101]. In order to be able to compare the two programs it was necessary to employ the closest possible calculation framework.

The first step was to define the geometry to be studied and the models that the programs under study use to model the injection moulding process. In this case the Navier-Stokes equations were used, considering the polymer melt as a compressible fluid throughout the process, as equation for the viscosity was used the Cross-WLF equation and as equation of state it was used the modified Tait equation. The evolution of the phase distribution was calculated through an advective transport equation, in both programs.

The material chosen for this study was the GPPS – Styron 678 Americas Styrenics, for which all the relevant parameters were identified. In order to be able to compare the results obtained with Moldex3D® with those obtained with the open-source program the ideal scenario would be to have used the same type of mesh, as it was not possible the objective was to use a similar number of cells. The meshes were obtained with, more or less, the same number of cells, but when analysing the meshes, although the Moldex3D® gave the information that the number of cells would be equal to the one of openInjMoldSim, the same did not seem to verify. When the Moldex3D® results were analysed all the meshes seemed to correspond to coarser meshes when compared to the ones in openInjMoldSim.

The next step was to define the same initial and boundary conditions in both programs. Since Moldex3D® provides several simplifications for the program to minimize the user requirements, to the process of understanding the actual conditions that were being employed was more complex. As an example, when defining the temperature conditions on the walls, at an initial attempt the same conditions were not imposed in both programs. In some cases the conditions in Moldex3D® were defined in a different manner that the ones for openInjMoldSim. For instance, to set the same inlet flow rate it was necessary to firstly run a simulation in Moldex3D®, convert the flow rate results in velocity and implement it in openInjMoldSim. Moreover, in the case of openInjMoldSim the properties of the air also had to be defined, and although being required for the calculations, they are absent in Moldex3D®.

The simulations were run and in terms of performance the difference between the two programs was noticeable, with runs being much slower in openInjMoldSim, on average being 18x slower. When performing the first simulation in openInjMoldSim, it was found that for the simulation to run there was a need to have some material inside the cavity at the beginning of the calculation. To assume similar conditions, it was required to identify the minimum amount of material that allowed to successful run in openInjMoldSim. After overpassing this difficulty, the obtained results were analysed and compared, starting by the evolution of the flow front, which was very similar between the two programs. Then, for the temperature and velocity distributions, several incoherent results were obtained for Moldex3D®. As the most representative results of what happens in the injection moulding process were those of the cross-section, these were the results used to compare the two programs.

The results of temperature were very similar, the temperature near the walls had a similar variation and the values were approximate between the two programs, with the refinement of the mesh the temperature variations near the walls approached. In the case of the velocity results, these were approaching with the refinement of the mesh. With Mesh 1 the relative difference was 20%, 15% for Mesh 2 and finally on Mesh 3 the relative difference of the maximum velocity reached was 4%. To analyse the pressure results were evaluated the pressure contours throughout the part, these were more or less linear, the difference that there was between programs was the maximum value of pressure reached. In the case of Mesh 1 the difference between the results of openInjMoldSim and Moldex3D® was notorious, the pressure drop at the same instant for openInjMoldSim corresponded to more than double the pressure drop obtained in the Moldex3D®. This was explained by the filling phase that was in a very final phase and since the material was poorly compressible there was an exponential increase of the pressure drop, for the most refined meshes this difference, which is so conspicuous, has ceased to exist and the relative error decreases.

In general, the results obtained through the open-source program appeared to be approaching the results obtained through the commercial program widely used by industry, Moldex3D®. In conclusion, with this project it was possible to achieve the objective of comparing both programs and to find a methodology that would allow this comparison. It has not yet been possible to get completely converged results so it has not yet been possible to validate the open-source program, but it was a relevant contribution towards this goal.

2. FUTURE WORK

As it was not yet possible to validate the solver and it presents some less good points, it would be important as future work:

- Compare the results of both programs using the same methodology, but this time use converged meshes instead of meshes with the same number of cells.
- Validate the solver by comparing it with experimental results.
- When looking at the results and the analysis of the mesh there are issues that should be clarified, in this sense to make a rigorous comparison of the solvers performance.

References

[1] G. Cardeal, ‘Energy Consumption Modelling in the Injection Moulding Industry’, pp. 1–10, 2016.

[2] P. Kennedy and R. Zheng, *Flow Analysis of Injection Molds*, 2nd ed. Hanser Publishers, 2013.

[3] T. A. Osswald, L.-S. Turng, and P. J. Gramann, Eds., *Injection Molding Handbook*. Hanser Publishers, 2002.

[4] T. Osswald and J. Hernández-Ortiz, *Polymer Processing: modeling and simulation*, 1st ed. Carl Hanser Verlag GmbH & Co. Kg, 2006.

[5] J. Shoemaker, ‘Moldflow Design Guide : A Resource for Plastics Engineers’, *Moldflow Des. Guid.*, p. 326, 2006.

[6] E. N. Peters, ‘Chapter 11 Plastics : Thermoplastics, Thermosets and Elastomers’, in *Handbook of Materials Selection*, vol. 10.1002/97, M. Kutz, Ed. John Wiley & Sons, 2002.

[7] Z. Huamin, Ed., *Computer Modeling For Injection Molding: Simulation, Optimization, and Control*. John Wiley & Sons, 2013.

[8] M. P. Stevens, ‘Chemistry of industrial polymers’, 2016. [Online]. Available: https://www.britannica.com/topic/industrial-polymer-chemistry-468716. [Accessed: 10-Apr-2018].

[9] F. A. Morrison, *Understanding Rheology*. Oxford University Press, Inc., 2001.

[10] D. V. Rosato, D. V. Rosato, and M. G. Rosato, Eds., *Injection Molding Handbook*, 3rd ed. Kluwer Academic Publishers.

[11] T. G. Mezger, *The Rheology Handbook: For users of rotational and oscillatory rheometers.*, 2nd ed. Hannover: Vincentz Network, 2006.

[12] ‘Basics of viscometry’. [Online]. Available: https://wiki.anton-paar.com/en/basic-of-viscometry/. [Accessed: 01-Jan-2019].

[13] S. Kazemian, A. Prasad, and B. B. K. Huat, ‘Review of Newtonian and non-Newtonian fluids behaviour in the context of grouts’, *Geotech. Asp. Undergr. Constr. Soft Gr.*, pp. 321–326, 2012.

[14] R. P. Chhabra, ‘Non-Newtonian fluids: An introduction’, *Rheol. Complex Fluids*, pp. 3–34, 2010.

[15] G. Ligia and R. Deodato, *Physicochemical Behavior and Supramolecular Organization of Polymers*, vol. 1. Springer Netherlands, 2009.

[16] ‘More solutions to sticky problems: a guide to getting more from your Brookfield Viscometer & Rheometer from the world leader in viscosity measurement and control’. AMETEK Brookfield, Inc., 2017.

[17] T. Osswald and N. Rudolph, *Polymer Rheology - Fundamentals and Applications*. Munich: Carl Hanser Verlag, 2015.

[18] C. A. Hieber and H. H. Chiang, ‘Shear rate dependence modeling of polymer melt viscosity’, *Polym. Eng. Sci.*, vol. 32, no. 14, p. 931, 1992.

[19] J. A. Forrest, K. Dalnoki-Veress, J. R. Stevens, and J. R. Dutcher, ‘Effect of free surfaces on the glass transition temperature of thin polymer films’, *Phys. Rev. Lett.*, vol. 77, no. 10, pp. 2002–2005, 1996.

[20] W. N. Dos Santos, J. A. De Sousa, and R. Gregorio, ‘Thermal conductivity behaviour of polymers around glass transition and crystalline melting temperatures’, *Polym. Test.*, vol. 32, no. 5, pp. 987–994, 2013.

[21] ‘Thermal Conductivity’. [Online]. Available: http://www.nde-ed.org/EducationResources/CommunityCollege/Materials/Physical_Chemical/ThermalConductivity.htm. [Accessed: 24-Apr-2018].

[22] I. L. Ngo, S. Jeon, and C. Byon, ‘Thermal conductivity of transparent and flexible polymers containing fillers: A literature review’, *Int. J. Heat Mass Transf.*, vol. 98, pp. 219–226, 2016.

[23] J. Wen, ‘Heat Capacities of Polymers’, in *Physical Properties of Polymers Handbook*, J. Mark, Ed. Springer, New York, NY, 2007.

[24] R. H. Shoulberg, ‘The thermal diffusivity of polymer melts’, *J. Appl. Polym. Sci.*, vol. 7, no. 40, pp. 1597–1611, 1963.

[25] H. Rees, *Understanding Injection Mold Design*. Carl Hanser Verlag GmbH & Co. Kg, 2001.

[26] J. Wang, ‘PVT Properties of Polymers for Injection Molding’, in *Some Critical Issues for Injection Molding*, J. Wang, Ed. 2012.

[27] B. Kowalska, ‘Processing aspects of p-v-T relationship’, *Polimery*, vol. 51, pp. 862–865, 2006.

[28] M.-L. Wang, R.-Y. Chang, and C.-H. (David) Hsu, ‘Material Properties of Plastics’, in *Molding Simulation: Theory and Practice*, Carl Hanser Verlag GmbH & Co. Kg, 2018, pp. 19–54.

[29] ‘PVT Model’. [Online]. Available: http://support.moldex3d.com/r15/en/standardinjectionmolding_material_materialmodels_pvtmodel.html. [Accessed: 06-Jun-2018].

[30] E. Júnior, R. Soares, and N. Cardozo, ‘Analysis of equations of state for polymers’, vol. 25, no. 3, pp. 277–288, 2015.

[31] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics -The finite volume method*, 2nd ed. Pearson Education Limited, 2007.

[32] C. Johnson, *Numerical solutions of partial differential equations by the finite element method*. Dover publications, Inc., 2009.

[33] S. W. Kim and L. S. Turng, 'Developments of three-dimensional computer-aided engineering simulation for injection moulding', *Model. Simul. Mater. Sci. Eng.*, vol. 12, no. 3, p. S151, 2004.

[34] S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*. New York: McGraw-Hill, 1980.

[35] J. H. Ferziger and M. Perić, *Computational Methods for Fluids Dynamics*, 3rd ed. Springer, 2002.

[36] O. C. Zienkiewicz, *The Finite Element Method*. New York: McGraw-Hill, 1977.

[37] M. Rovito, 'Electromigration Reliability Issue in Interconnects for Three-Dimensional Integration Technologies'. [Online]. Available: http://www.iue.tuwien.ac.at/phd/rovitto/node62.html. [Accessed: 25-Sep-2018].

[38] C. A. Brebbia, J. C. F. Telles, and L. Wrobel, *Boundary Element Techniques: Theory and Applications in Engineering*. New York: Springer, 1984.

[39] P. K. Banerjee, *The Boundary Element Methods in Engineering*, 2nd ed. New York: McGraw-Hill, 1994.

[40] M. Costabel, 'Principles of boundary element methods', *Comput. Phys. Reports*, vol. 6, no. 1–6, pp. 243–274, 1987.

[41] F. Moukalled, L. Mangani, and M. Darwish, *The Finite Volume Method in Computational Fluid Dynamics*, vol. 113. 2016.

[42] R. Pantani, V. Speranza, and G. Titomanlio, 'Thirty years of modeling of injection molding. A brief review of the contribution of unisa code to the field', *Int. Polym. Process.*, vol. 31, no. 5, pp. 655–663, 2016.

[43] G. D. Gilmore and R. S. Spencer, 'Role of Pressure, Temperature, and Time in Injection Molding Process', *Mod. Plast.*, vol. 37, no. 8, pp. 143–151, 1950.

[44] R. S. Spencer and G. D. Gilmore, 'Residual Strains in Injection Molded Polysterene', *Mod. Plast.*, vol. 28, no. 4, pp. 97–105, 1950.

[45] R. S. Spencer and G. D. Gilmore, 'Equation of State for High Polymers', *J. Appl. Phys.*, vol. 21, no. 6, pp. 523–526, 1950.

[46] G. D. Gilmore and R. S. Spencer, 'Photographic Study of Polymer Cycle in Injection Molding', *Mod. Plast.*, vol. 28, no. 8, pp. 117–124, 1951.

[47] P. Thyregod, 'Modelling and Monitoring in Injection Molding', Technical University of Denmark, 2001.

[48] R. L. Ballman and H. L. Toor, 'Injection Molding: Rheological Interpretation', *Mod. Plast.*, vol. 31,

no. 1–2, pp. 105–111, 1959.

[49] R. L. Ballman and H. L. Toor, ‘Orientation in Injection Molding’, *Mod. Plast.*, vol. 38, no. 2, pp. 113–122, 1960.

[50] G. B. Jackson and R. L. Ballman, ‘Effect of Orientation on Physical Properties of Injection Moldings’, *Soc. Plast. Eng. - J.*, vol. 16, no. 10, pp. 1147–1152, 1960.

[51] H. L. Toor, R. L. Ballman, and L. Cooper, ‘Predicting Mold Flow by Electronic Computer’, *Mod. Plast.*, vol. 38, no. 4, pp. 117–120, 1960.

[52] D. H. Harry and R. G. Parrot, ‘Numerical simulation of injection mold filling’, *Polym. Eng. Sci.*, vol. 10, no. 4, pp. 209–214, 1970.

[53] H. A. Lord and G. Williams, ‘Mold-filling studies for the injection molding of thermoplastic materials. Part II: The transition flow of plastic materials in the cavities of injection molding dies.’, *Polym. Eng. Sci.*, vol. 15, no. 8, pp. 569–582, 1975.

[54] P. Thienel and G. Menges, ‘Mathematical and Experimental-Determination of Temperature, Velocity, and Pressure Fields in Flat Molds During Filling Process in Injection-Molding of Thermoplastics’, *Polym. Eng. Sci.*, vol. 18, no. 4, pp. 314–320, 1978.

[55] G. Williams and H. A. Lord, ‘Mold-filling studies for the injection molding of thermoplastic materials. Part I: The flow of plastic materials in hot and cold-walled circular channels’, *Polym. Eng. Sci.*, vol. 15, no. 8, pp. 553–568, 1975.

[56] R. E. Nunn and R. T. Fenner, ‘Flow and heat transfer in the nozzle of an injection molding machine’, *Polym. Eng. Sci.*, vol. 17, no. 11, pp. 811–818, 1977.

[57] M. R. Kamal and S. Kenig, ‘The injection molding of thermoplastics. Part I: Theoretical model’, *Polym. Eng. Sci.*, vol. 12, no. 4, pp. 294–301, 1972.

[58] M. R. Kamal and S. Kenig, ‘The injection molding of thermoplastics. Part II: Experimental test of the model’, *Polym. Eng. Sci.*, vol. 12, no. 4, pp. 302–308, 1972.

[59] P. C. Wu, C. F. Huang, and C. G. Gogos, ‘Simulation of the moldfilling process’, *Polym. Eng. Sci.*, vol. 14, no. 3, pp. 223–230, 1974.

[60] J. F. Stevenson, ‘A simplified method for analyzing mold filling dynamics. Part I: Theory’, *Polym. Eng. Sci.*, vol. 18, no. 7, pp. 577–582, 1978.

[61] J. F. Stevenson and W. Chuck, ‘A simplified method for analyzing mold filling dynamics. Part II: Extensions and comparisons with experiment’, *Polym. Eng. Sci.*, vol. 19, no. 12, pp. 849–857, 1979.

[62] E. C. Bernhardt, Ed., *CAE Computer Aided Engineering for Injection Molding*. Hanser Publishers,

1983.

[63] Z. Tadmor, E. Broyer, and C. Gutfinger, ‘Flow analysis network (FAN) - A method for solving flow problems in polymer processing’, *Polym. Eng. Sci.*, vol. 14, no. 9, pp. 660–665, 1974.

[64] E. Broyer, C. Gutfinger, and Z. Tadmor, ‘A theoretical model for the cavity filling process in injection molding’, *J. Rheol. (N. Y. N. Y).*, vol. 19, p. 423, 1975.

[65] W. L. Krueger and Z. Tadmor, ‘Injection molding into a rectangular cavity with inserts.’, *Polym. Eng. Sci.*, vol. 20, no. 6, pp. 426–431, 1980.

[66] M. R. Kamal, Y. Kuo, and P. H. Doan, ‘The injection molding behavior of thermoplastics in thin rectangular cavities.’, *Polym. Eng. Sci.*, vol. 15, no. 12, pp. 863–868, 1975.

[67] Y. Kuo and M. R. Kamal, ‘The fluid mechanics and heat transfer of injection mold filling of thermoplastic materials’, *AIChE J.*, vol. 22, no. 4, pp. 661–669, 1976.

[68] C. A. Hieber and S. F. Shen, ‘A finite-element/finite-difference simulation of the injection-molding filling process’, *J. Nonnewton. Fluid Mech.*, vol. 7, no. 1, pp. 1–32, 1980.

[69] S. F. Shen, ‘Simulation of polymeric flows in the injection moulding process’, *Int. J. Numer. Methods Fluids*, vol. 4, no. 2, pp. 171–183, 1984.

[70] C. L. Tucker, *Fundamentals of Computer Modeling for Polymer Processing*. Hanser: Munich, Germany, 1989.

[71] D. Cardozo, ‘Three models of the 3D filling simulation for injection molding: A brief review’, *J. Reinf. Plast. Compos.*, vol. 27, no. 18, pp. 1963–1974, 2008.

[72] C. Fernandes, A. J. Pontes, J. C. Viana, and A. Gaspar-Cunha, ‘Modeling and Optimization of the Injection-Molding Process: A Review’, *Adv. Polym. Technol.*, vol. 37, no. 2, pp. 429–449, 2018.

[73] C. A. Hieber and S. F. Shen, ‘Flow analysis of the nonisothermal 2-dimensional filling process in injection-molding’, *Isr. J. Technol.*, vol. 16, no. 5–6, pp. 248–254, 1978.

[74] V. W. Wang, C. A. Hieber, and K. K. Wang, ‘Filling of an arbitrary three dimensional thin cavity’, *J. Polym. Eng.*, vol. 7, pp. 21–45, 1986.

[75] K. Himasekhar, L. Lottey, and K. K. Wang, ‘CAE of mold cooling in injection molding using a three-dimension numerical solution’, *ASME J. Eng. Ind.*, vol. 114, pp. 213–221, 1992.

[76] N. Santhanam, *Analysis of Residual Stresses and Post- Molding Deformation in Injection-Molded Components*. Ithaca, New York: Cornell University, 1992.

[77] M. Gupta and K. K. Wang, ‘Effect of processing conditions on fiber orientation in short-fiber-reinforced injection-molded composites’, in *SPE Annual Technical Conference - ANTEC*, 1993, pp. 2290–2295.

[78] N. Santhanam, H. H. Chiang, K. Himasekhar, P. Tuschak, and K. K. Wang, ‘Postmolding and load-induced deformation analysis of plastic parts in the injection molding process’, *Adv. Polym. Technol.*, vol. 11, no. 2, pp. 77–89, 1991.

[79] X. Jin and Z. Zhao, ‘Shrinkage-warpage prediction method for injection-molded fiber-filled polymer composite parts’, in *SPE Annual Technical Conference—ANTEC*, 1998, pp. 614–618.

[80] L. S. Turng, V. W. Wang, and K. K. Wang, ‘Numerical simulation of the coinjection molding process’, *J. Eng. Mater. Technol.*, vol. 115, no. 1, pp. 48–53, 1993.

[81] L. S. Turng, ‘Development and application of CAE technology for the gas-assisted injection molding process’, *Adv. Polym. Technol.*, vol. 14, no. 1, pp. 1–13, 1995.

[82] S. Han, K. K. Wang, and D. L. Crouthamel, ‘Wire-sweep study using an industrial semiconductor-chip-encapsulation operation’, *J. Electron. Packag.*, vol. 119, no. 4, pp. 247–254, 1997.

[83] S. C. Chen, Y. C. Chen, N. T. Cheng, and E. Al, ‘Simulation of injection-compression mold-filling process’, *Int. Commun. Heat Mass Transf.*, vol. 25, no. 7, pp. 907–917, 1998.

[84] K. M. Pillai, ‘Modeling the unsaturated flow in liquid composite molding processes: A review and some thoughts’, *J. Compos. Mater.*, vol. 38, no. 23, pp. 2097–2118, 2004.

[85] H. H. Chiang, C. A. Hieber, and K. K. Wang, ‘A unified simulation of the filling and postfilling stages in injection molding. Part I: Formulation’, *Polym. Eng. Sci.*, vol. 31, no. 2, pp. 116–124, 1991.

[86] H. H. Chiang, C. A. Hieber, and K. K. Wang, ‘A unified simulation of the filling and postfilling stages in injection molding. Part II: Experimental Verification’, *Polym. Eng. Sci.*, vol. 31, no. 2, pp. 125–139, 1991.

[87] H. Zhou, Y. Zhang, and D. Li, ‘Injection moulding simulation of filling and post-filling stages based on a three-dimensional surface model’, *Proc. Inst. Mech. Eng. Part B – J. Eng. Manuf.*, vol. 215, no. 10, pp. 1459–1463, 2001.

[88] H. Zhou and D. Li, ‘A numerical simulation of the filling stage in injection molding based on a surface model’, *Adv. Polym. Technol.*, vol. 20, no. 2, pp. 125–131, 2001.

[89] H. Zhou and D. Li, ‘Computer filling simulation of injection molding based on 3D surface model’, *Polym. - Plast. Technol. Eng.*, vol. 41, no. 1, pp. 91–102, 2002.

[90] D. M. Gao, K. T. Nguyen, J. F. Hétu, and et al, ‘Modeling of industrial polymer processes: Injection molding and blow molding’, *Adv. Perform. Mater.*, vol. 5, no. 1–2, pp. 43–64, 1998.

[91] R. Chang and W. Yang, ‘Numerical simulation of mold filling in injection molding using a three-dimensional finite volume approach’, *Int. J. Numer. Methods Fluids*, vol. 37, no. 2, pp. 125–148,

2001.

[92] V. Rajupalem, K. Talwar, and C. Friedl, 'Three-dimensional simulation of the injection molding process', in *SPE Annual Technical Conference—ANTEC*, 1997, pp. 670–673.

[93] S. W. Kim and L. S. Turng, 'Three-dimensional numerical simulation of injection molding filling of optical lens and multiscale geometry using finite element method', *Polym. Eng. Sci.*, vol. 46, no. 9, pp. 1263–1274, 2006.

[94] G. Haagh and F. N. Van De Vosse, 'Simulation of three-dimensional polymer mould filling processes using a pseudo-concentration method', *Int. J. Numer. Methods Fluids*, vol. 28, no. 9, pp. 1355–1369, 1998.

[95] R. Y. Chang, W. H. Yang, S. J. Hwang, and F. Su, 'Three-dimensional modeling of mold filling in microelectronics encapsulation process', *Polym. Eng. Sci.*, vol. 27, no. 1, pp. 200–209, 2004.

[96] J. Zhou and L. S. Turng, 'Three-dimensional numerical simulation of injection mold filling with a finite-volume method and parallel computing', *Adv. Polym. Technol.*, vol. 25, no. 4, pp. 247–258, 2006.

[97] B. J. Araújo, J. C. F. Teixeira, A. M. Cunha, and C. P. T. Groth, 'Parallel three-dimensional simulation of the injection molding process', *Int. J. Numer. Methods Fluids*, vol. 59, no. 7, pp. 801–815, 2008.

[98] K. C. Estacio and N. Mangiavacchi, 'Simplified model for mould filling simulations using CVFEM and unstructured meshes', *Commun. Numer. Methods Eng.*, vol. 23, no. 5, pp. 345–361, 2007.

[99] S. L. Jiang, Z. G. Wang, G. F. Zhou, and et al, 'An implicit control-volume finite element method and its time step strategies for injection molding simulation', *Comput. Chem. Eng.*, vol. 31, no. 11, pp. 1407–1418, 2007.

[100] Pichelin and Coupez, 'Finite element solution of the 3D mold filling problem for viscous incompressible fluid', *Comput. Methods Appl. Mech. Eng.*, vol. 163, pp. 359–371, 1998.

[101] 'About Moldex3D'. [Online]. Available: https://www.moldex3d.com/en/about/. [Accessed: 05-Jun-2018].

[102] 'Moldex3D Products Overview'. [Online]. Available: https://www.moldex3d.com/en/products/software/moldex3d/. [Accessed: 05-Jun-2018].

[103] 'Mathematical Models and Assumptions'. [Online]. Available: http://support.moldex3d.com/r15/en/standardinjectionmolding_flow_reference_mathematical modelsandassumptions.html. [Accessed: 06-Jun-2018].

[104] 'Viscosity model'. [Online]. Available:

http://support.moldex3d.com/r15/en/standardinjectionmolding_material_materialmodels_viscositymodel(thermoplasticonly).html. [Accessed: 06-Jun-2018].

[105] ‘Viscoelasticity model’. [Online]. Available: http://support.moldex3d.com/r15/en/standardinjectionmolding_material_materialmodels_viscoelasticitymodel(thermoplasticonly).html. [Accessed: 06-Jun-2018].

[106] ‘Features’. [Online]. Available: https://www.autodesk.com/products/moldflow/features. [Accessed: 06-Jun-2018].

[107] ‘About the Differences between 3D and Dual Domain/midplane Analysis’, 2016. [Online]. Available: https://knowledge.autodesk.com/support/moldflow-insight/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/3PP-MDF-INSFUND-ASCENT/files/GUID-13421821-C017-4118-89E8-35E49443D628-htm.html. [Accessed: 05-Jun-2018].

[108] ‘Fast fill analysis (Concept)’, 2017. [Online]. Available: https://knowledge.autodesk.com/support/moldflow-insight/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/MoldflowInsight/files/GUID-1F96E63E-7425-4A9B-8E72-217167883B53-htm.html. [Accessed: 05-Jun-2018].

[109] ‘Simulation models’, 2017. [Online]. Available: https://knowledge.autodesk.com/support/moldflow-insight/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/MoldflowInsight/files/GUID-EC0F471C-A2C5-4461-9C83-B62B85936166-htm.html. [Accessed: 05-Jun-2018].

[110] K. Krebelj and J. Turk, ‘Opensource injection molding simulation. A solver for OpenFOAM’. [Online]. Available: https://github.com/krebeljk/openInjMoldSim. [Accessed: 05-Jun-2018].

[111] N. Mole, K. Krebelj, and B. Štok, ‘Validacija numeričnega modela za simulacijo brizganja plastike s pomočjo izmerjene evolucije tlaka v votlini’, in *SLOVENSKO DRUŠTVO ZA MEHANIKO SREČANJE KUHLJEVI DNEVI 2017*, 2017, pp. 73–80.

[112] F. Ospald, ‘Numerical Simulation of Injection Molding using OpenFOAM’, *Appl. Math. Mech.*, vol. 14, no. 1, pp. 673–674, 2014.

[113] B. J. Araújo, ‘Injection molding’. [Online]. Available: https://www.sites.google.com/site/billyaraujo/other-software/injection-molding. [Accessed: 07-Jun-2018].

[114] B. Fisa and M. Rahmani, ‘Weldline strength in injection molded glass fiber-reinforced polypropylene’, *Polym. Eng. Sci.*, vol. 31, no. 18, pp. 1330–1336, 1991.

[115] X. Zhuang, J. Ouyang, W. Li, and Y. Li, ‘Three-dimensional simulations of non-isothermal transient

flow and flow-induced stresses during the viscoelastic fluid filling process', *Int. J. Heat Mass Transf.*, vol. 104, pp. 374–391, 2017.

[116] S.-C. Xue, R. I. Tanner, and N. Phan-Thien, 'Three-dimensional numerical simulations of viscoelastic flows - predictability and accuracy', *Comput. Methods Appl. Mech. Engrg.*, no. 180, pp. 305–331, 1999.

[117] H. Zhou, B. Yan, and Y. Zhang, '3D filling simulation of injection molding based on the PG method', *J. Mater. Process. Technol.*, vol. 204, pp. 475–480, 2008.

[118] R. I. T. Rong Zheng and X.-J. Fan, *Injection Molding : Integration of theory and modeling methods*. 2011.

[119] 'Polymer properties database - Polystyrenes (GPPS, HIPS, EPS, SBR, SBS, ABS)'. [Online]. Available: https://polymerdatabase.com/polymer classes/Polystyrene type.html. [Accessed: 12-Dec-2018].

[120] 'Use Boundary Layer Mesh (BLM) for Complex Geometries to Ensure Accuracy of Mold-Filling Analyses', 2015. [Online]. Available: https://www.moldex3d.com/en/blog/top-story/use-boundary-layer-mesh-blm-for-complex-geometries-to-ensure-accuracy-of-mold-filling-analyses/. [Accessed: 31-Jan-2019].