

Cooling of Photovoltaic Microchips

a project presented to
The Faculty of the Department of Aerospace Engineering
San José State University

in partial fulfillment of the requirements for the degree
Master of Science in Aerospace Engineering

by

Zachariah Eme

May 2023

approved by

Yawo Ezunkpe
Faculty Advisor



© 2022
Zachariah Eme
ALL RIGHTS RESERVED

Abstract

This thesis outlines the process of developing a computational approach for photovoltaic cell cooling design. As electronic components shrink, effective heat maintenance becomes more difficult. As excessive temperatures within a system can cause early degradation and failure, developing an effective method of heat transport has the potential to provide significant longevity benefits to computer chips. As a primary source of power in space applications, photovoltaics on long-term missions need to maintain functionality. This research project develops a computational methodology for cooling channel design by comparing heat transfer coefficients from standard correlations compared to CFD results for a microfluidics chip cooling. This is done in hopes of providing a basis for new design work for microfluidics cooling.

Acknowledgments

I would like to express my gratitude to Dr Yawo Ezunkpe, for his direction and experience in the world of CFD. You provided tremendous knowledge in the pursuit of this project. I would also like to thank Benjamin Ard, who as an undergraduate is unable to have his name on the title of this document, but our work together was the basis for this project. I hope to publish in a conference or journal with you soon.

Table of Contents

Abstract	iii
Acknowledgments	iv
List of Tables	vii
List of Figures	viii
Nomenclature	x
1 Introduction	1
1.1 Motivation	1
1.2 Literature Review	1
1.2.1 Microfluidics	1
1.2.2 Commercial Software Evaluation	3
1.2.3 Computational Fluid Dynamics Considerations	4
1.2.4 Other on Chip Cooling Work	5
1.3 Objective	6
1.4 Methodology	6
2 Mathematical Model	7
2.1 Fluid Mechanics	7
2.1.1 Navier-Stokes Equations	7
2.1.2 Conservation of Energy	8
3 Computational Approximation	10
3.1 Finite Volume	10
3.2 Modern CFD Algorithms	10
3.3 Sources of Error	11
3.3.1 Modelling Error	11
3.3.2 Discretization Error	11
3.3.3 Convergence Error	11
3.4 Stability	11
4 Heat Modeling	13
4.1 Conduction	13
4.2 Convection	14
4.2.1 Forced Convection	14
5 Experimental Set Up	16

5.1	Three Dimensional Model and Meshing	17
5.2	Mesh	18
5.3	Setup	18
6	Results	24
6.1	Sample Heat Transfer 1D Comparison Calculation	24
6.1.1	Initial Non-Dimensional Number Calculations	24
6.1.2	Heat Transfer Coefficient Using Laminar Nusselt Number Correlation	25
6.1.3	Heat Transfer Coefficient Using Turbulent Nusselt Number Correlation	25
6.1.4	Heat Transfer Coefficient Using Constant Surface Temperature Approximation	26
6.1.5	Heat Transfer Coefficient Using Constant Heat Flux Assumption	26
6.2	Effects of Varying Mass Flow	26
6.3	Effects of Varying Surface Roughness	27
6.4	Effects of Varying Flow Inlet Temperature	28
6.5	Surface Temperature vs Bulk Outlet Temperature	29
6.6	Comparison of CFD Heat Transfer Coefficient With Nusselt Number Calculations	30
7	Discussion and Analysis	35
7.1	Proposed Model Predicted Average Silicon Maximum Temperatures vs CFD Results	36
8	Conclusions and Recommendations for Further Research	39
	References	40
A	Matlab Code	44
B	Full Table of Heat Transfer Coefficient Results	50

List of Tables

5.1	Typical Solar Panel Layer Thickness [1]	17
6.1	Sample calculation properties	24
6.2	Sample CFD results	24
6.3	Heat Transfer Coefficient Comparison, Values in $\frac{W}{m^2K}$	34
B.1	Heat Transfer Coefficient Comparison, Values in $\frac{W}{m^2K}$	50

List of Figures

1.1	Velocity profile in a closed channel [2].	2
4.1	Composite slab resistance model [3]	13
5.1	General solar panel structure [4]	16
5.2	EVA properties [5]	16
5.3	Glass properties	17
5.4	Silicon properties	17
5.5	Side profile of chip	18
5.6	Mesh convergence of maximum temperature reached in silicon chip	18
5.7	Mesh convergence of average temperature reached in silicon chip	19
5.8	Mesh convergence of average convective heat transfer coefficient	19
5.9	Mass inlet setup	20
5.10	Mass outlet setup	21
5.11	Convergence control	22
5.12	Nested iteration setup	23
5.13	Table of run points	23
6.1	Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 289K	27
6.2	Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 291K	27
6.3	Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 293K	28
6.4	Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 295K	28
6.5	Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 297K	29
6.6	Effect of changing mass flow rate on convective heat transfer coefficient as compared to 0.0009 kg/s reference case	29
6.7	Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 289K	30
6.8	Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 291K	30
6.9	Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 293K	31
6.10	Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 295K	31
6.11	Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 297K	32

6.12	Effect of changing surface roughness on convective heat transfer coefficient as compared to zero roughness case	32
6.13	Effect of changing fluid inlet temperature on convective heat transfer coefficient	33
6.14	Difference between maximum interface temperature and bulk outflow temperature	33
7.1	Solar panel thermal resistance model	36
7.2	Error in maximum temperature prediction by proposed model	38

Nomenclature

CFD	Computational Fluid Dynamics
Re	Reynolds Number
Pr	Prandtl Number
Nu	Nusselt Number
Ca	Capillary Number
μ	viscosity
ρ	density
ν	momentum diffusivity
α	thermal diffusivity
P	pressure
h	enthalpy
SIMPLE	Semi-Implicit Method for Pressure-Linked Equations
PISO	Pressure Implicit with Splitting of Operator
PIMPLE	a combination of SIMPLE and PISO
PDES	partial differential equations

Chapter 1: Introduction

1.1 Motivation

Photovoltaics are used frequently in satellites that orbit the Earth because a solar panel and battery system provide better weight characteristics together than a battery alone, an important factor in the cost of launches [6]. As commercial accessibility to space opens up, the market for small-scale satellites has been growing- due to their higher affordability [7]. These smaller satellites have less space between components and as a result, are more likely to have heat carryover between sensitive components. This heat bleed can lead to decreased lifespans [8] or, as in the case with the James Webb telescope, make temperatures too high to get accurate readings [9]. Most cooling of these designs is performed via either conduction or radiation, but for cases where more heat removal is necessary, heat transfer via a cooling flow can provide a significant benefit. By performing analysis and design on fluid-flow cooling in spacecraft solar panels, more information can be discovered on the feasibility of these systems and potentially contribute to future design work.

1.2 Literature Review

1.2.1 Microfluidics

Fluid dynamics can be broken down into different regimes based on Reynolds numbers, or as is the case with microfluidics, the dimensions of flow. In general, microfluidics is considered to be the region of flow where at least one dimension of a system is below 500 micrometers [10]. Due to the small scale channels, the surface area is much larger than the fluid volume, which allows for quick heat transfer and a potential solution to "hot-spots" that develop due to non-uniform power distribution [11].

Microfluidics, despite its vastly different scale than classic fluid problems, can still be described by the Navier-Stokes equations. In the control volume of the fluid, it is expected that the mass, momentum, and energy will all be conserved, and as such traditional numerical methods for computational fluid dynamics should apply in the solution of heat transfer in microfluidic channels [12].

Starting from the basic principles of fluid mechanics, the viscous shear stress in Newtonian fluids is characterized by the equation:

$$\tau = \mu \frac{\delta u}{\delta y} \tag{1.1}$$

where μ is the viscosity of the fluid, and u is the velocity of the fluid. In fluid mechanics, a common assumption is the no-slip condition- this dictates that fluid near

a solid boundary sticks to that boundary. For channel flow with non-moving walls, this implies no velocity at the boundaries. As flow develops in the closed channel, a parabolic velocity curve is expected to result. This parabola has a greater strain rate near the boundary, and thus a greater shear stress.

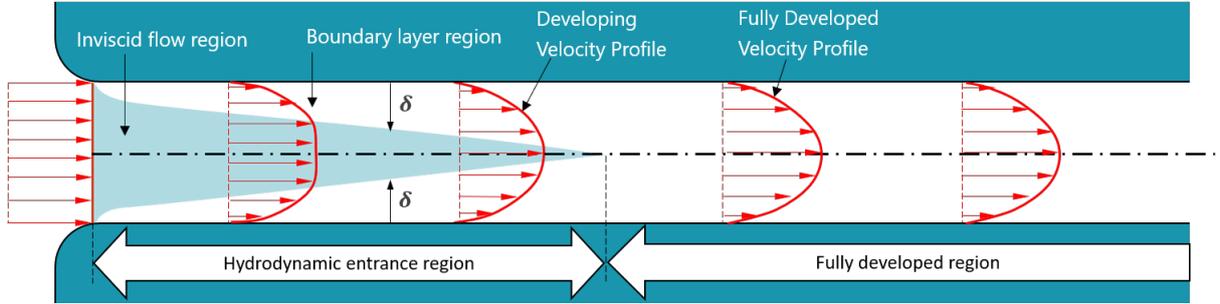


Figure 1.1: Velocity profile in a closed channel [2].

A method of measuring the expected viscous forces against the inertial forces in a flow is the non-dimensional Reynolds number. All fluid analysis start with an evaluation of the Reynolds number to help get a sense of the expected flow characteristics. The Reynolds number is calculated as

$$Re = \frac{\rho UL}{\mu}. \quad (1.2)$$

where ρ is the density of the fluid, and L is the characteristic length of the fluid. In the case of closed channel flow, the characteristic length is the hydraulic diameter. As the diameter of the channel decreases, so does the Reynolds number, and therefore the viscous forces are expected to be more dominant in determining fluid motion. A Reynolds number of under 1500 [13] is generally considered to be a laminar flow, and for microfluidics, the expected Reynolds number is far below this value. This means that laminar flow and dominant viscous forces tend to characterize the fluid motion, resulting in an increased dependence on diffusive mixing for microfluidics devices.

Another important non-dimensional number to help characterize the flow is the Prandtl number,

$$Pr = \frac{\nu}{\alpha}. \quad (1.3)$$

where ν is the momentum diffusivity and α is the thermal diffusivity of the fluid. This dictates the relative thickness of the thermal boundary layer as compared to the momentum boundary layer. A smaller Prandtl number means quick heat diffusivity as compared with momentum and a thicker thermal boundary layer.

Another non-dimensional number relating to heat transfer is the Nusselt number,

$$Nu = \frac{h}{k/L} \quad (1.4)$$

where h is the convective heat transfer coefficient, k is the conductive heat transfer coefficient, and L is the characteristic length. The Nusselt number provides a methodology to compare the convective and conductive heat transfer in the flow, which is a beneficial way to evaluate expected heat transfer in a system.

The Peclet number is another tool for evaluating heat transfer in a flow. In general, it represents the ratio of advection to diffusion of any physical quantity in the flow. For heat transfer,

$$Pe = \frac{u}{\alpha/L} \quad (1.5)$$

where u is the flow velocity, α is the thermal diffusivity of the fluid, and L is the characteristic length.

The volume of fluid flow scales in a cylinder as

$$V = \pi r^2 L \quad (1.6)$$

and the surface area scales as

$$A = 2\pi r L \quad (1.7)$$

The ratio of volume to surface area scales as

$$\frac{V}{A} \propto r \quad (1.8)$$

As r decreases, the ratio of surface area to volume increases. Because microfluidics have a very low characteristic radius, the surface area is quite large as compared to the volume. This means surface tension is expected to have a large impact on fluid behaviour. A common measure of the relationship between the impact of viscous effects and surface tension is the Capillary number.

$$Ca = \frac{u \times \mu}{\gamma} \quad (1.9)$$

Computational Fluid Dynamics is a commonly applied tool for evaluating microfluidic flows in ways that are typically infeasible for experimental results. In free surface flows, a particular method known as the volume-of-fluid has been developed to conserve volume in high surface tension environments. For the purposes of this project, the flow is bounded by non-porous walls and is assumed to be single phase. This means that traditional computational fluid models should apply, but special care should be applied to make sure diffusive heat transfer is upheld.

The full derivation of equations driving fluid behavior is present in the Mathematical Model chapter, and the approximations and sources of error as a result from conversion into a numerical model can be found in the Computational Approximation chapter.

1.2.2 Commercial Software Evaluation

Commercial CFD software is used frequently in engineering science, and many tools are available to the average consumer. Each of these tools has benefits for different applications, and an essential part of any fluid modelling project is choosing a software that offers the accuracy and controllability necessary to resolve the projects regime of flow. For the benefit of this student project, three different commercial software offerings will be evaluated for their benefits in solving microfluidic problems: ANSYS FLUENT, OpenFOAM, and SOLIDWORKS. Access to FLUENT and SOLIDWORKS is provided by San Jose State University, and OpenFOAM is a free, open source software.

An ability to provide a stochastic heat input is of particular relevance to this project, as is general accuracy for microfluidics solution. Speed of solution would be of added benefit for the design stage of the project, but is the least important factor.

ANSYS FLUENT is a finite-volume method to solve the Navier-Stokes equations. It has the ability to simulate coupled physics, and has a graphical interface that provides lots of data about the fluid volume and makes user setup easy. In a validation of microfluidic chip flow mixing from 2021, the error of a two dimensional simulation against a full experimental test bed was 15.01% [14]. While this is not precise enough to constitute final design of a system, preliminary results of a design program as well as the general trends associated with changing variables of the design can be captured. The authors of the paper also astutely point out that the atmosphere of the experiment is unable to be as finely controlled as the computational simulation, and as such some of the error can be contributed to this variation. In single phase models such as the one proposed in the project, it would be anticipated for the results to be more accurate, as volume-of-fluid methods (as well as other interface determining methods) can lead to significant inaccuracies [15].

ANSYS FLUENT has the ability to collect time statistics, which allow for the collection of mean and root mean square values for a time dependent flow. For a stochastic process, this can help with post processing the results. The solver formulation is implicit [16], which means that all time steps are stable - this is of significant benefit for any time dependent solution.

OpenFOAM as an open source software offers many benefits. First and foremost, as a C++ based architecture, extra packages can be written by the end user to hook into the solver [17]. This includes but is not limited to visualization methods, solver techniques, and boundary conditions. The benefit of this flexibility is that a user who needs a specific methodology that is not included can tailor the program to their own needs. A large number of paid and free extensions to OpenFOAM are available, but are not required for solutions of microfluidics problems.

The packaging of several different user configurable solvers into OpenFOAM make it a good choice for microfluidics and any computational fluid problem. The difference between a good CFD solution and a bad one is dependent on proper solver, mesh, and boundary condition selection; the more customization and user control available, the more opportunities exist for a useful solution.

SOLIDWORKS has full integration between its own robust CAD software and its fluid modelling. It also has several fluid and solid material properties directly incorporated /citesolidworksForCFD such as silicon. This makes design around on chip cooling easier, with conjugate heat transfer a direct design point for this software. It also contains methodology for estimating boundary layer properties when there is not efficient mesh sizing, while letting a full Navier-Stokes solution find the properties when the mesh is sufficiently fine. The incompressible solver for SOLIDWORKS uses an operator splitting technique that is based around the SIMPLE algorithm, which shall be explained more in depth later in this paper. The coordination between the embedded CAD software and the fluids solution makes Solidworks excellent for parametric design studies.

1.2.3 Computational Fluid Dynamics Considerations

Computational fluid dynamics is a direct extension of the infinitesimal volumetric principles behind the formulation of Navier-Stokes. Most commercial methods involve

a finite volume method: a method where an area is discretized into a set of smaller volumes for the use of solving complex partial differential equations [18]. In the case of fluid dynamics, the partial differential equations are some form of the Navier-Stokes equations. To ease the computational load of solving these highly coupled equations, simplifying assumptions are typically applied. These simplifying assumptions are tied to the physical processes behind the Navier-Stokes equations, and as such are done on a case by case basis for individual problems [19], [20].

Special care must be placed on proper discretization of the problem. The truncation error in a numerical method is always tied to the sizing of a mesh- this leads to the big O notation [21]. The order of a method (otherwise known as O) is the relationship of truncation error to mesh sizing. As the mesh becomes finer, the truncation error of the method is expected to decrease proportionally to O. The mesh also needs to be several times smaller than the smallest scale waveform to provide reasonable representative data [22].

For any method that marches forwards in time, the numerical stability must also be considered. To have a stable numerical method, error should not grow between time steps [23]. Unstable numerical methods tend to result in unacceptable solutions that do not follow physical laws. To find the stability constraints of finite volume approaches, a Von Neuman analysis can be applied as Lauritzen did in 2006 [24].

Accurately modeling a fluid's interaction with the world around it requires proper handling of boundary conditions. These boundary conditions are additional constraints on a solution such that a unique solution should occur. In general, boundary conditions can be broken down into Dirichlet (a specified value of the function), Neumann (a specified normal derivative of a function), Robin (a linear combination of function values and specified normal derivative), or some mixture of the three [25].

Another important factor for modelling fluids is the initial condition of the fluid. A well defined problem needs the initial state of the fluid in the regime of interest. Both time marching and constant time iteration methods require an initial solution to iterate from. Initial conditions in a discretized region should match the initial conditions expected in the real world scenario.

1.2.4 Other on Chip Cooling Work

Microfluidics in heat removal for electronics has been typically constrained to computer chips, but the lessons applied there can be of great usefulness to any heat transfer project on a small scale. A methodology called microchannel heat sink is very similar to the idea proposed in this project. These typically focus on material type, surface treatment, and channel configuration for use in optimization of microchannels [26]. Studies on cross sectional shape have suggested that rectangular cross sections outperform triangular and trapezoidal channels (The effect of geometrical parameters on heat transfer characteristics of microchannels heat sink with different shapes, Gunnassegaran et al). Disturbing the heat boundary layer has also been shown to increase net heat transfer into microchannels by up to 44% (Resonance of the thermal boundary layer adjacent to an isothermally heated vertical surface). Increasing thermal conductivity of the material surround the channel has also been shown to increase heat performance [27].

1.3 Objective

The objective of this project is to develop a CFD solution for heat removal from solar panels by means of fluid flow. Heat fluctuations in solar panels are non-predictable, but an average can be fairly easily determined across any specific time frame. Because of this, any heat-removal needs to be able to work in both higher than average and lower than average heat flux conditions. A regime of fluid inlet temperatures, mass flow rates, and surface roughness will be used to provide a comparison to standard convective heat transfer coefficients to aid in future design work.

1.4 Methodology

To achieve the established goals of this project, an analysis of the flow regime as well as commercially available solvers will be performed. Then, a mathematical model will be derived from first principles, with all assumptions listed and explained. Based on the desired fidelity of the model, the mesh sizing and time step will be set, with an analysis as to the stability and relevance of this particular solving method. Solutions will focus on steady-state heat input and will be compared to hand calculations using known fluid properties. This will provide a baseline of expected heat transfer coefficients and will end in a proposed approach to calculate convective heat transfer coefficients by hand.

Chapter 2: Mathematical Model

2.1 Fluid Mechanics

2.1.1 Navier-Stokes Equations

As mentioned in the literature review, the Navier-Stokes partial differential equations describe the properties of a viscous fluid's motion. These highly coupled equations are developed by mathematically modeling the conserved quantities of mass and momentum.

In a fluid volume, the continuity equation states that the accumulation of mass within the volume is equal to the net mass flow into the volume (mass flow in minus mass flow out). This equation is represented as:

$$\frac{D\rho}{Dt} + \nabla \cdot (\rho \vec{V}) = 0 \quad (2.1)$$

where ρ is fluid density, t is time, and \vec{V} is the flow velocity field. Liquids are generally considered to be incompressible [28], and the continuity equation simplifies to:

$$\frac{D\rho}{Dt} + \rho (\nabla \cdot \vec{V}) = 0 \quad (2.2)$$

The conservation of momentum in a viscous fluid is represented by three equations representing momentum in the x , y , and z directions. It can be thought of as an application of Newton's Second Law

$$F = ma \quad (2.3)$$

(where F is force, m is mass, and a is acceleration) to a generalized infinitesimal fluid volume. Because then limit of this volume goes to zero, its possible to write this as

$$\rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) = f_{body} + f_{surface} \quad (2.4)$$

where u is the x direction velocity, v is the y direction velocity, and w is the z direction velocity. The left-hand side of this equation is the density of the fluid multiplied by the acceleration in the x direction. In this application, the only body force is the force of gravity on the fluid,

$$f_{body} = \rho g \quad (2.5)$$

The surface forces in a fluid can be broken into 2 categories: the force due to pressure, and the shear force due to viscous effects in the fluid.

$$f_{surface} = -\frac{\partial P}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \quad (2.6)$$

Putting this all together, we get this for the x direction conservation of momentum equation.

$$\rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) = -\frac{\partial P}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \quad (2.7)$$

In vector form, the conservation of momentum equation is represented by:

$$\rho \frac{D\vec{U}}{Dt} = -\nabla P + \mu \Delta \vec{V} + \vec{F} \quad (2.8)$$

2.1.2 Conservation of Energy

Another conserved property in fluids is energy. From applying the first law of thermodynamics [29] to fluid passing through a control volume, the following energy equation results:

$$\frac{\delta E_t}{\delta t} = -\nabla \cdot E_t \vec{V} \frac{\delta Q}{\delta t} - \nabla \cdot \vec{q} + \rho \vec{f} \cdot \vec{V} + \nabla \cdot (\vec{\Pi}_{ij} \cdot \vec{V}) [23], [30] \quad (2.9)$$

where E_t is the total internal energy and is mathematically represented via

$$E_t = \rho \left(e + \frac{\vec{V}^2}{2} + E_{potential} + E_{chemical} \right) \quad (2.10)$$

In Eq. 2.9, the left hand side is the accumulation term, and the right hand side is made of the rates of energy loss via convection, heat production, heat loss via conduction, the work done by body forces, and the work done by surface forces.

In Cartesian form [23], this becomes

$$\frac{\delta E_t}{\delta t} = \frac{\delta Q}{\delta t} + \rho(f_x u + f_y v + f_z w) \quad (2.11)$$

$$- \quad (2.12)$$

$$\frac{\delta}{\delta x} (E_t u + p u - u \tau_{xx} - v \tau_{xy} - w \tau_{xz} + q_x) \quad (2.13)$$

$$- \quad (2.14)$$

$$\frac{\delta}{\delta y} (E_t v + p v - u \tau_{xy} - v \tau_{yy} - w \tau_{yz} + q_y) \quad (2.15)$$

$$- \quad (2.16)$$

$$\frac{\delta}{\delta z} (E_t w + p w - u \tau_{xz} - v \tau_{zy} - w \tau_{zz} + q_z) \quad (2.17)$$

If internal energy and kinetic energy are considered to be the only significant forms of energy in Eq. 2.10, Eq. 2.9 becomes

$$\rho \frac{Dh}{Dt} = \frac{Dp}{Dt} + \frac{\delta q}{\delta t} - \nabla \cdot \vec{q} + \Phi \quad (2.18)$$

where Φ is the dissipation function written using Einstein notation

$$\Phi = \tau_{ij} \frac{\delta u_i}{\delta x_j} [31] \quad (2.19)$$

and τ_{ij} is the viscous stress tensor. A full accounting of this stress tensor is not included in this work, but is widely available in fluid textbooks and other resources.

The incompressibility of the flow was assumed earlier, which simplifies Eq. 2.18 to

$$\rho \frac{Dh}{Dt} = \frac{\delta Q}{\delta t} - \nabla \cdot \vec{q} + \Phi \quad (2.20)$$

This equation closes the loop on heat transfer in the system.

Chapter 3: Computational Approximation

3.1 Finite Volume

Because the Navier-Stokes equations have no known direct solution, computational fluid dynamics relies on discretizing the fluid flow into control volumes. The governing conservation equations are then integrated across the discrete elements to transform them into algebraic equations. The resulting system of algebraic equations is then solved. The flux of a conserved property leaving a control volume must be equal to the flux entering the face-attached volume in the finite volume solution, meaning that finite volume is a strictly conservative approach. It also allows for unstructured meshes- making it good at discretizing irregular volumes [32].

Fluid properties are stored at the center of the control volume, but in order to determine how these properties behave and are transported, Gauss' Theorem is used [32].

$$\int_V (\nabla \cdot \vec{u}) dV = \oint_S \vec{u} \cdot \vec{n} d\vec{S} = \sum_i^{numFace} \vec{u}_i^f \cdot \vec{S}_i^f \quad (3.1)$$

Where $numFace$ is the number of faces in the control volume, \vec{u}_i^f is the flux through the face, and \vec{S}_i^f is the face surface. Simply stated, Gauss' Theorem (or divergence theorem) says that the net flux of a property is equivalent to the sum of sources and sinks of the field in its volume. The surface integral can be replaced by a summation across the control volume faces and can be approximated by the face's flux at the centroid.

The interpolation scheme used dictates how the the face-centered value is approximated. Linear and upwind approaches are what are most commonly used. These are a major contributor to the accuracy of any finite volume method.

3.2 Modern CFD Algorithms

Modern commercial CFD algorithms tend to use one of three methods of solution for the Navier-Stokes equations. These algorithms are known by PISO (Pressure Implicit with Splitting of Operator), SIMPLE (Semi-Implicit Method for Pressure-Linked Equations), and PIMPLE (a blended approach of PISO and SIMPLE). To explain the solutions of commercial CFD codes, it is worthwhile to know what they are actually doing in the process of solutions.

The general approach for SIMPLE and PISO is the same, the only difference is where the corrector loop takes place. SIMPLE has an outer corrector and is mostly only suitable for steady-state problems, whereas PISO has an inner corrector loop and is used for transient problems. The general flow of these approaches are:

1. Solve the matrix momentum equation for the velocity field
2. Use the velocity field to compute a residual
3. Use the residual in the continuity equation to calculate a new pressure field
4. Use the new pressure field and the residual to correct the velocity field

The SIMPLE algorithm repeats steps 1 to 4 until convergence criteria is met, whereas the PISO algorithm repeats steps 2 to 4 for around two or three steps. PIMPLE is an approach that utilizes both the outer and inner corrective loops from SIMPLE and PISO. This means that for each time step, the outer corrective loop is ran until convergence criteria is met, and for each outer loop run, you would have 2 to 3 inner corrective loop runs.

3.3 Sources of Error

3.3.1 Modelling Error

Modelling errors are a direct result of using incorrect PDEs to model the flow. Navier-Stokes (and the conservation of energy) are good models for single phase, non-reacting fluid flow. This leads to negligible modelling errors. [33]

3.3.2 Discretization Error

Another source of error is the discretization process. As mentioned in Section 3.1, the PDEs are transformed into a linear system of equations. This process leads to discretization error [34]. Discretization error can be decreased by increasing the number of control volumes and by increasing the order of interpolation. For this project, a second order scheme was used for interpolation. This means the discretization error decreases as a square of the mesh size- a higher mesh convergence rate than first order, but without some of the robustness issues higher order codes can have in unstructured meshes.

3.3.3 Convergence Error

The third error type, convergence errors are a result of incomplete solutions to the system of linear equations [34]. The solution to the systems of linear equations are iterative, and converge to some level of tolerance. This tolerance bounds the convergence error. This convergence error is used to approximate the truncation error- because the true solution is unknown, the difference between the analytical solution and the numerical solution is unknown.

3.4 Stability

When solving transient flows, time step selections and time derivative methods can lead to unstable solutions- numerical approximations that tend towards infinity. Time dependent PDEs are approximated by either implicit or explicit methods. Explicit methods take in a current value and step it forwards via some time step. The simplest of these is Forwards Euler's method:

$$Y_{k+1} = Y_K + \Delta t f(t_k, Y_k) \quad (3.2)$$

This is a first order method [35]. The simplest implicit method is Backwards Euler's method:

$$Y_{k+1} = Y_K + \Delta t f(t_k, Y_{k+1}) \quad (3.3)$$

This is also first order, but it is more computationally challenging to solve than the Forwards Euler [35]. Implicit methods are generally more stable than explicit methods which means larger time steps can be used [36]. The increased computational expense for each iteration is less significant than the increase in stability, and therefore implicit algorithms are more typically used.

A method for evaluating the time step of any time integration scheme is the CFL (Courant-Frederichs-Lewy) number [37]. For an n-dimensional problem space,

$$CFL = \Delta t \left(\sum_{i=1}^n \frac{u_{x_i}}{\Delta x_i} \right) \quad (3.4)$$

where u_{x_i} is the magnitude of the velocity in the x_i direction, and Δx_i is the spacial step in the x_i direction. For an explicit method, $CFL < 1$ to maintain stability, but for an implicit method no such limits need to be applied for stability. Implicit methods will benefit from a low CFL to minimize numerical temporal diffusion, meaning that a limit of $CFL < 1$ can be useful for accuracy [38], but is not strictly necessary for stability.

Chapter 4: Heat Modeling

Heat transfer has three modes: conduction, radiation, and convection [39]. Conduction occurs as a result of the diffusion of heat. This occurs within both solids and fluids, and heat is transferred from high temperatures to low temperatures. Radiation is the direct result of electromagnetic energy transfer through space, and convection is the direct result of a moving fluid transporting heat away from a body. The major forms of heat transfer in this model are conduction and convection, with radiation being approximated as a heat generation upon the top surface of the solar panel.

Heat models that are solved by hand very frequently rely on a resistance model—a model that measures heat transfer as overcoming thermal resistance. In this model, heat flow rate is modeled in the same fashion as current and temperature differences correspond to voltage drop across the thermal resistor [3]. An example of a composite slab resistance model is shown in Figure 4.1.

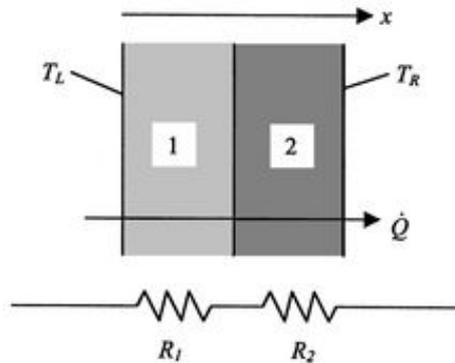


Figure 4.1: Composite slab resistance model [3]

4.1 Conduction

Conduction is primarily a function of temperature gradient, the conductivity of a material, and the area it is conducting through. This takes the form of:

$$q = -h\nabla T[40] \quad (4.1)$$

When taken down to a one dimensional model, this results in:

$$q = -kA \frac{\Delta T}{\Delta X}[40] \quad (4.2)$$

Where A is the cross sectional area, k is the thermal conductivity, and q is the rate of heat transfer. In terms of the thermal resistance model, the resistance takes the form:

$$R_{conductive} = \frac{\Delta X}{kA} [3] \quad (4.3)$$

4.2 Convection

Convection can be broken down into two categories: forced and free convection. Free convection is the result of natural behaviour of a fluid, such as buoyant properties and forced convection relies on a pump or some other device moving the flow. Forced convection is what is used in the process of internally cooling a chip.

4.2.1 Forced Convection

Forced convection, and convection in general is a very complex topic that is dependant upon fluid temperature, velocity, surface roughness, surface temperature and much more. To measure the rate of energy absorption by conduction, the following equation can be used if a temperature independent specific heat capacity is assumed.

$$q = \dot{m}C_p\Delta T [41] \quad (4.4)$$

The rate of heat flux into the fluid can be found via:

$$q_s'' = h(T_s - T_m) [41] \quad (4.5)$$

Where h is the convective heat transfer coefficient, T_s is the temperature of the solid surface, and T_m is the mean temperature of the fluid. The 1D differential equation that represents the heat gradient due to convection is:

$$\frac{dT_m}{dx} = \frac{P_{pipe}}{\dot{m}C_p} h(T_s - T_m) [41] \quad (4.6)$$

Where P_{pipe} is the perimeter of the interaction between the forced flow and the solid. A common assumption to make this differential equation more solvable by hand is a constant surface temperature such that:

$$\frac{dT_m}{dx} = -\frac{d(T_s - T_m)}{dx} = -\frac{d(\Delta T)}{dx} [41] \quad (4.7)$$

Separation of variables and integrating from $x = 0$ to $x = L$ results in:

$$\ln\left(\frac{\Delta T_0}{\Delta T_1}\right) = \frac{-P_{pipe}L}{\dot{m}C_p} \bar{h} [41] \quad (4.8)$$

which can be simplified to:

$$\frac{T_s - T_{m,0}}{T_s - T_{m,1}} = e^{\frac{-P_{pipe}L}{\dot{m}C_p} \bar{h}} [41] \quad (4.9)$$

where \bar{h} is the spacial average convective heat transfer coefficient, $T_{m,0}$ is the mean fluid temperature entering the region of forced convection, and $T_{m,1}$ is the mean fluid temperature exiting.

To find \bar{h} , either the temperatures must be known, or a correlation for the average heat transfer coefficient must be used. For forced convection, the Nusselt number is

commonly used to find the convective heat transfer coefficient. For non-turbulent flow and a constant surface temperature, the following approximation is commonly used:

$$\overline{Nu_d} = 3.66 + \frac{0.0668G_{ZD}}{1 + 0.04G_{ZD}^{2/3}} [41] \quad (4.10)$$

Where G_{ZD} is the Graetz number, found by

$$G_{ZD} = \frac{D}{x} Re_D Pr [41] \quad (4.11)$$

For turbulent flow, the Nusselt number correlation commonly used is the Dittus-Boelter Equation:

$$\overline{Nu_d} = 0.023 Re_D^{4/5} Pr^{0.3} [41] \quad (4.12)$$

These will be compared to CFD simulation results of a microfluidics system to determine their validity. These values for \bar{h} can be used in the thermal resistance model as well, with the form:

$$R_{convective} = \frac{1}{hA} \quad (4.13)$$

Chapter 5: Experimental Set Up

Due to SOLIDWORKS robust integration between CAD and CFD modelling as well as its extensive library of material properties, it was selected as the CFD software used for this project. To model a solar panels heat transfer characteristics, first the structure of the solar panel needs to be evaluated.

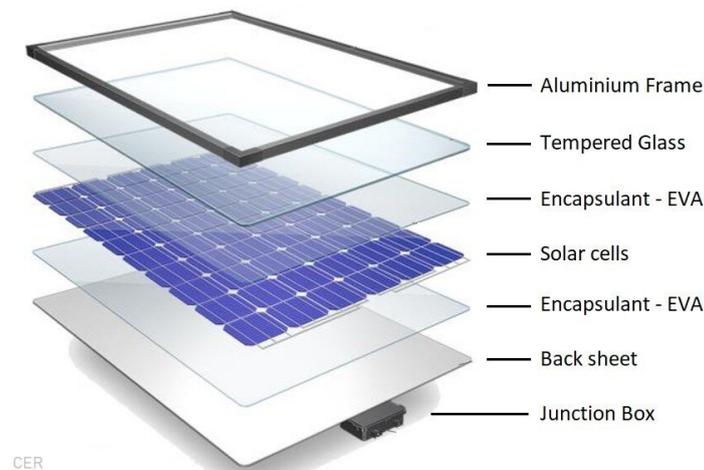


Figure 5.1: General solar panel structure [4]

Shown in Figure 5.1, a solar panel is made of layers of glass, ethylene vinyl acetate (EVA), and solar cells. For this model, the solar cells will be considered as solid silicon, the glass as the standard properties set out in SOLIDWORKS, and the EVA is user defined based on properties from [5]. The user defined properties are as follows in Figure 5.2:

Property	Value
Name	Ethylene-Vinyl-Acetate (EVA)
Comments	
Density	961 kg/m ³
Specific heat	1400 J/(kg*K)
Conductivity type	Isotropic
Thermal conductivity	0.34 W/(m*K)
Electrical conductivity	Dielectric
Radiation properties	<input type="checkbox"/>
<input checked="" type="checkbox"/> Melting temperature	<input checked="" type="checkbox"/>
Temperature	368.15 K

Figure 5.2: EVA properties [5]

The properties used for glass are shown in Figure 5.3

Property	Value
Name	Glass
Comments	
Density	2600 kg/m ³
Specific heat	670 J/(kg*K)
Conductivity type	Isotropic
Thermal conductivity	(Table)
Electrical conductivity	Dielectric
Radiation properties	<input type="checkbox"/>
☐ Melting temperature	<input checked="" type="checkbox"/>
Temperature	1073.15 K

Figure 5.3: Glass properties

and the properties used for silicon are shown in Figure 5.4. These properties are included within SOLIDWORKS.

Property	Value
Name	Silicon
Comments	
Density	2330 kg/m ³
Specific heat	(Table)
Conductivity type	Isotropic
Thermal conductivity	(Table)
Electrical conductivity	Dielectric
Radiation properties	<input type="checkbox"/>
☐ Melting temperature	<input checked="" type="checkbox"/>
Temperature	1688.2 K

Figure 5.4: Silicon properties

The aluminum frame will be ignored, and water channels will be ran through the EVA backing underneath the solar cells. The thickness of the individual solar panel layers is shown in Table 5.1.

Table 5.1: Typical Solar Panel Layer Thickness [1]

Material	Thickness (mm)
Glass	3
EVA	0.45
Solar Cell	0.18
EVA	0.60

The average value of the EVA backing is 0.45 mm, but up to 0.15 mm extra can be added without significant changes in solar panel performance. The model will be set up as shown in Figure 5.5, with layers of glass, EVA, Silicon, and EVA with holes in the EVA for overall chip cooling.

5.1 Three Dimensional Model and Meshing

A model was built with the average thickness defined for each layer in Table 5.1. The chip was defined to be 100mm by 100mm, and 200 holes of 0.2mm radius were spaced equally through the bottom EVA layer.

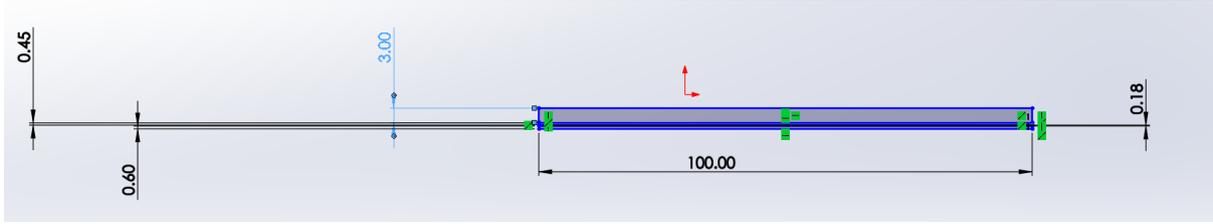


Figure 5.5: Side profile of chip

5.2 Mesh

The meshing process was performed using Solidworks automatic meshing tool. A mesh convergence study was performed. The results of the mesh convergence study and the final mesh chosen are displayed in Figures 5.6, 5.7, and 5.8.

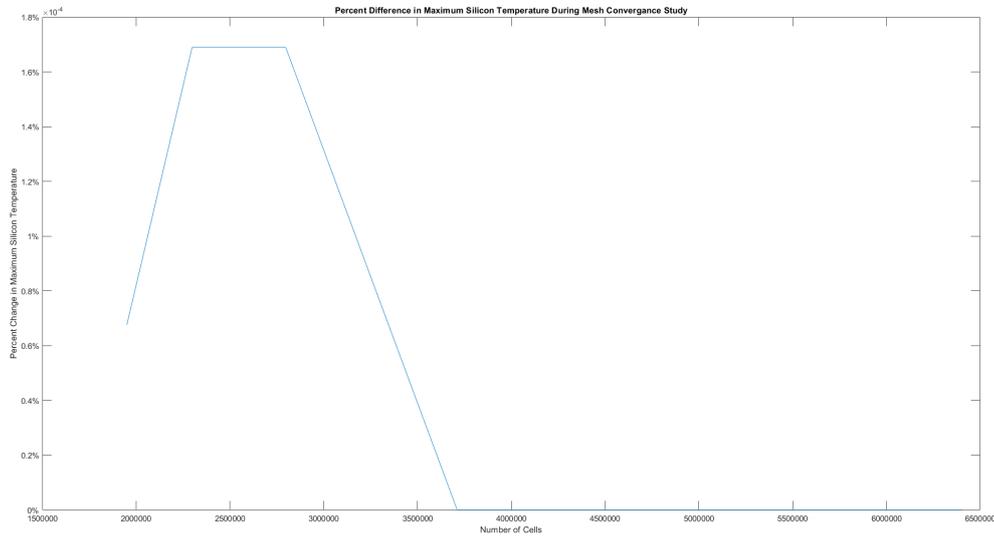


Figure 5.6: Mesh convergence of maximum temperature reached in silicon chip

This convergence study only has three points, but a doubling in mesh nodes led to under 1% change for the maximum temperature reached in the silicon, the average temperature of the silicon, and for the convective heat transfer coefficient.

5.3 Setup

In this study, surface roughness, fluid temperature, and fluid mass flow rate are varied. The fluid enters with a fully developed momentum profile and leaves at a static pressure of 1 atm.

The calculation is considered converged when the average temperature in the fluid varies by less than 0.01 Kelvin across half of what Solidworks calls a "travel." A "travel" is the number of iterations for information to be fully communicated downstream of a flow.

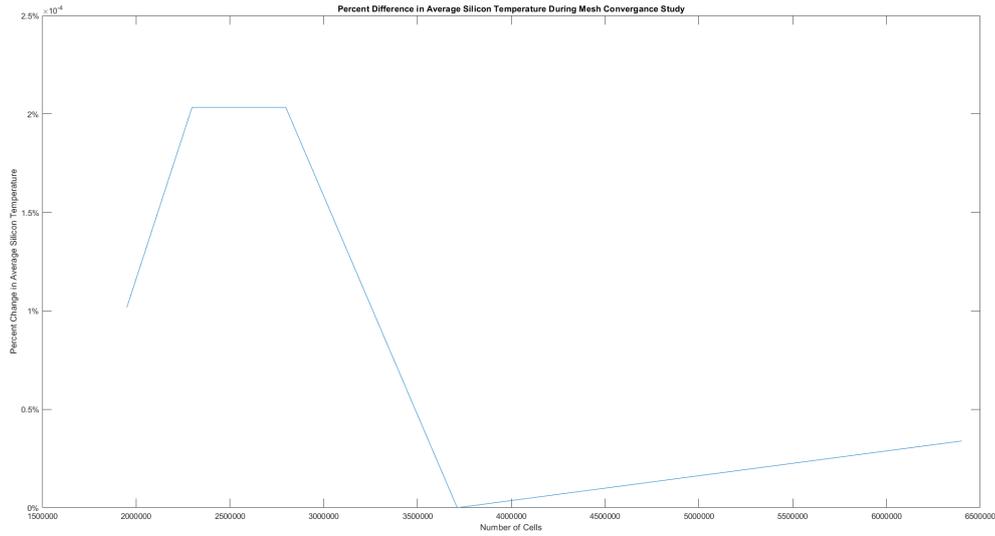


Figure 5.7: Mesh convergence of average temperature reached in silicon chip

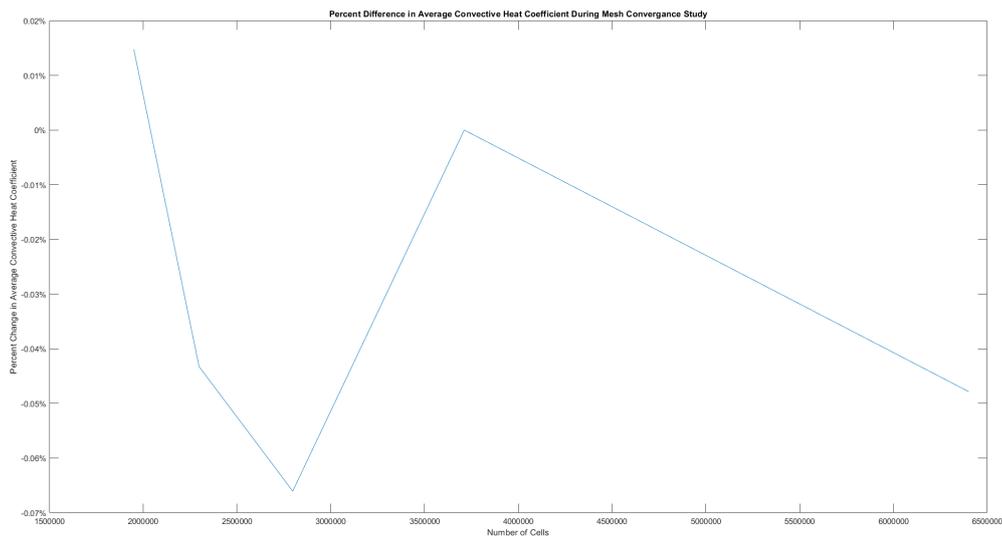


Figure 5.8: Mesh convergence of average convective heat transfer coefficient

No mesh refinement steps are taken during solution, and 12 nested iterations occur as this was found to provide the best overall solution speed per case.

The variable parameters are shown in Figure 5.13

This results in 60 total cases that will be compared against the analytical Nusselt number approximations.

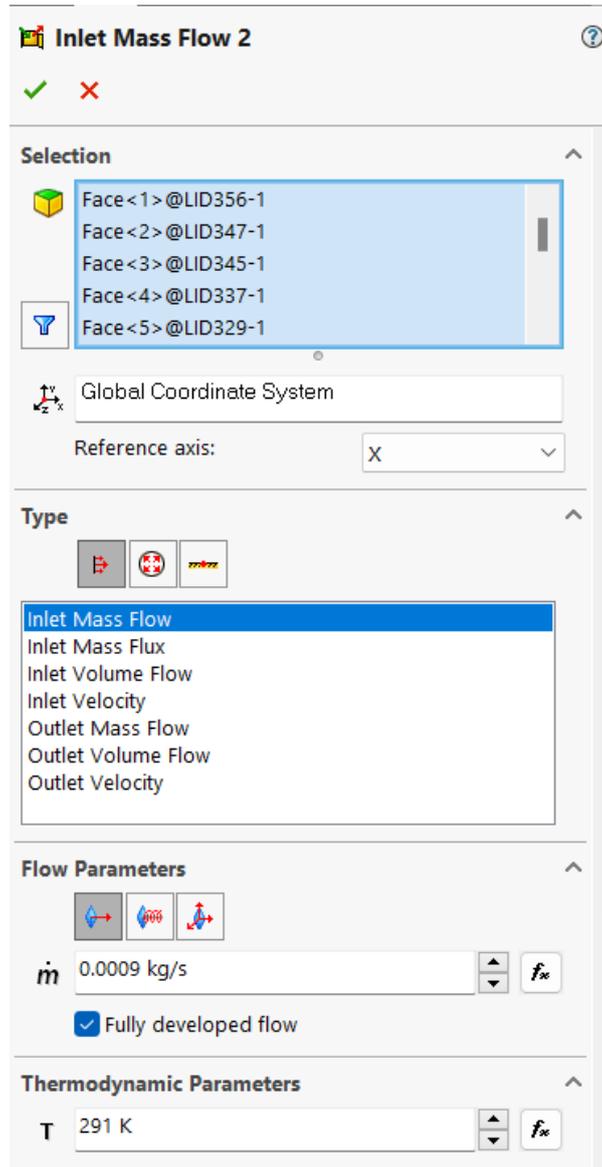


Figure 5.9: Mass inlet setup

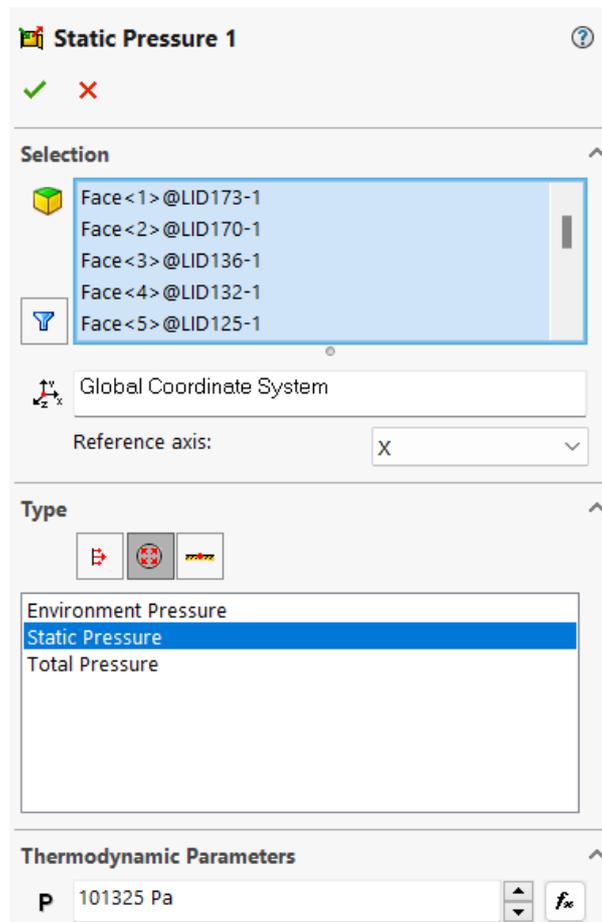


Figure 5.10: Mass outlet setup

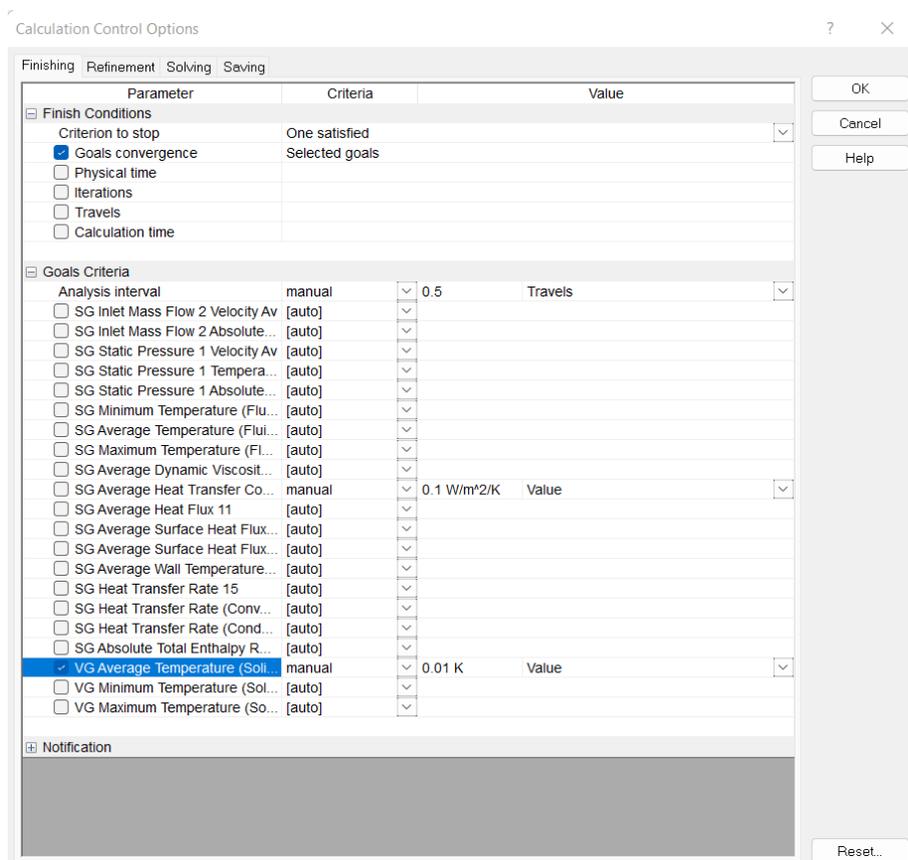


Figure 5.11: Convergence control

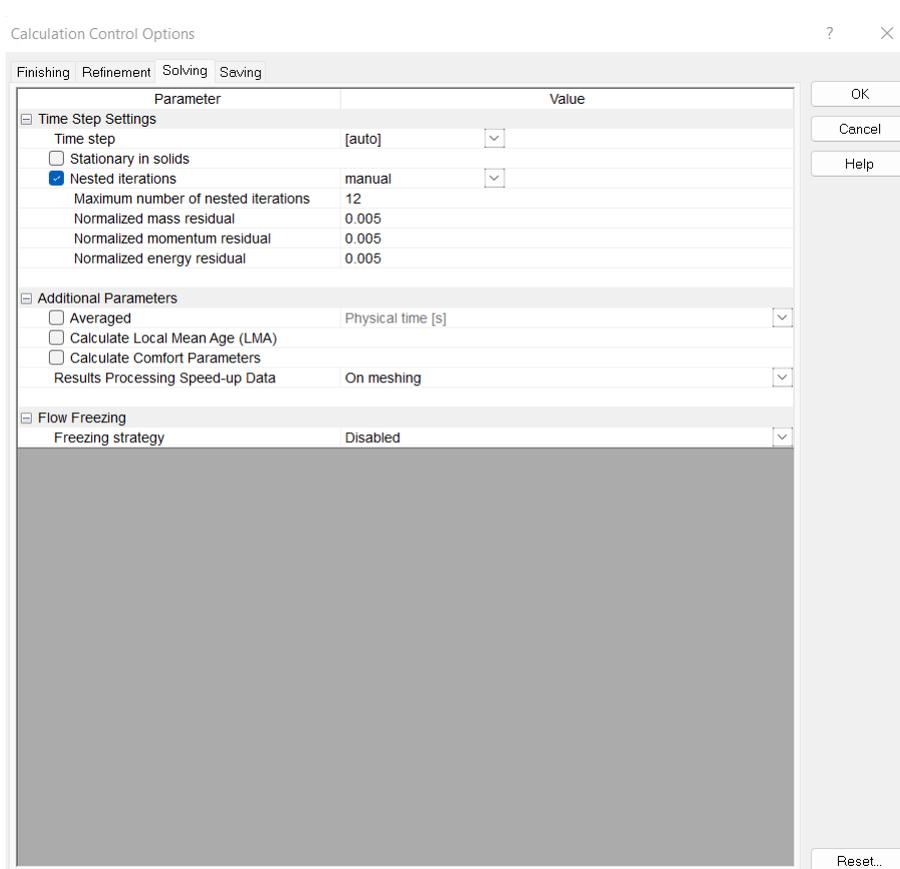


Figure 5.12: Nested iteration setup

Parameter	Current Value	Variation Type	#	Values
Roughness (Wall Conditions)	0 micrometer	Discrete Values	3	0, 0.1, 0.2
Pressure (Initial Conditions)	101325 Pa	Discrete Values	1	101325
Temperature (Initial Conditions)	293.2 K	Discrete Values	1	293.2
Initial solid temperature (Initial Conditions)	295 K	Discrete Values	1	295
Level of initial mesh (Automatic Mesh)	2	Discrete Values	1	2
Mass flow rate (Inlet Mass Flow 2)	0.001 kg/s	Discrete Values	4	0.0009, 0.001, 0.0011, 0.0012
Temperature (Inlet Mass Flow 2)	289 K	Discrete Values	5	289, 291, 293, 295, 297
Surface heat generation rate (SS Surface Heat Generation Rate 1)	1000 W/m ²	Discrete Values	1	1000

Figure 5.13: Table of run points

Chapter 6: Results

6.1 Sample Heat Transfer 1D Comparison Calculation

A sample calculation for the convective heat transfer coefficient, \bar{h} is shown to compare with the results of the first CFD case ran. This process was repeated for all other cases to show the differences between expected results and the results obtained by CFD. The comparison case has the properties listed in Table 6.1.

Table 6.1: Sample calculation properties

Property	Value
Roughness	0 micrometers
Mass Flow Rate	0.0009 kg/s
Fluid Inlet Temperature	289 K
Top Surface Heat Generation Rate	1000 W/m^2

The CFD results are listed in Table 6.2.

Table 6.2: Sample CFD results

Property	Value
Silicon Average Temperature	290.99 K
Silicon Minimum Temperature	289.83 K
Silicon Maximum Temperature	292.27 K
Inlet Mass Flow Average Velocity	0.0387 m/s
Fluid Outlet Temperature	291.65 K
Surface Average Temperature	290.40 K
Average Heat Transfer Coefficient	167.08 $W/(m^2K)$
Average Heat Flux	414.95 W/m^2

6.1.1 Initial Non-Dimensional Number Calculations

To find the heat transfer coefficient, the Reynolds number must first be found.

$$Re_D = \frac{\rho u_M D}{\mu} = \frac{u_M D}{\nu} \quad (6.1)$$

The fluid inlet temperature is 289 K, and the outlet temperature is 291.65 K. The average kinematic viscosity for water [42] in this temperature range is

$$\nu = 1.0756 * 10^{-6} (m^2/s) \quad (6.2)$$

The value for the Reynolds number is then

$$Re_D = \frac{0.0387 * 4 * 10^{-4}}{1.0756 * 10^{-6}} = 14.392 \quad (6.3)$$

The turbulent threshold for Reynolds numbers is considered to be around 2300, so the laminar Nusselt number calculations are expected to give a better result. Another important number for calculation of the Nusselt Number correlations is the Prandtl number. In Chapter 9 of *Microfluidics: Modelling, Mechanics and Mathematics*, author Bastian E. Rapp states: "Usually, the Prandtl number is assumed to be around 0.7 for gases and around 6.9 for water. [43]" Therefore, for the comparison cases ran,

$$Pr = 6.9 \quad (6.4)$$

6.1.2 Heat Transfer Coefficient Using Laminar Nusselt Number Correlation

The Laminar Nusselt Number correlation relies on the Graetz number. The Graetz number in this example case is found via

$$G_{ZD} = \frac{D}{x} Re_D Pr = \frac{4 * 10^{-4} (m)}{0.1 (m)} 14.392 * 6.9 = 0.39722 \quad (6.5)$$

Plugging in to the average Nusselt number correlation gives

$$\overline{Nu_d} = 3.66 + \frac{0.0668 * 0.39722}{1 + 0.04 * 0.39722^{2/3}} = 3.686 \quad (6.6)$$

The average heat transfer coefficient is then found via

$$Nu_D = \frac{\bar{h} D_h}{k} \quad (6.7)$$

where D_h is the hydraulic diameter. In a circular tube, the hydraulic diameter is the interior diameter of the tube. For water in this temperature range, $k = 0.59255 \frac{W}{mK}$ [44].

Plugging in gives

$$\bar{h} = \frac{Nu_D k}{D_h} = \frac{3.686 * 0.59255}{4 * 10^{-4}} = 5461 \frac{W}{m^2 K} \quad (6.8)$$

6.1.3 Heat Transfer Coefficient Using Turbulent Nusselt Number Correlation

The turbulent Nusselt number correlation is

$$\overline{Nu_d} = 0.023 Re_D^{4/5} Pr^{0.3} = 0.023 * 14.392^{4/5} * 6.9^{0.3} = 0.3466 \quad (6.9)$$

this gives a average heat transfer coefficient of

$$\bar{h} = \frac{Nu_D k}{D_h} = \frac{0.3466 * 0.59255}{4 * 10^{-4}} = 513.5 \frac{W}{m^2 K} \quad (6.10)$$

6.1.4 Heat Transfer Coefficient Using Constant Surface Temperature Approximation

The constant surface temperature 1 dimensional heat transfer problem results in the following equation

$$\frac{T_s - T_{m,0}}{T_s - T_{m,1}} = e^{\frac{-P_{pipe} L}{\dot{m} C_p} \bar{h}} \quad (6.11)$$

From the CFD results, $T_s = 290.40K$, $T_{m,0} = 289K$, $T_{m,1} = 291.65K$. The perimeter of the pipe is given by

$$P_{pipe} = \pi D = \pi * 4 * 10^{-4} = 0.001257(m) \quad (6.12)$$

and the heat capacity of the water is $4.182 \frac{J}{kgK}$ [43]. Plugging in gives

$$\frac{290.40 - 289}{290.40 - 291.65} = e^{\frac{-0.001257 * 0.1}{0.0009 * 4.182} \bar{h}} \quad (6.13)$$

this is clearly a non-physical result, and ends in a non-solvable equation. This shows that the constant surface temperature approximation does not work- as the temperature of the fluid leaving is higher than the average temperature.

6.1.5 Heat Transfer Coefficient Using Constant Heat Flux Assumption

For a constant heat flux under laminar conditions, the Nusselt number is given by

$$Nu_D = 4.36 \quad (6.14)$$

Solving this for the average convective heat transfer coefficient gives

$$\bar{h} = 6458.8 \frac{W}{m^2 K} \quad (6.15)$$

The result of this correlation does not match with the value found in the CFD solution.

6.2 Effects of Varying Mass Flow

For inflow at a temperature of 189K, it appears as though increasing mass flow rate decreases the heat transfer coefficient as seen in Figure 6.1. For an inlet temperature of 291K as shown in Figure 6.2, an increase in mass flow rate seems to increase the convective heat transfer coefficient and then decrease \bar{h} as the flow rate goes above a threshold of 0.0011 kg/s for the 0micrometer roughness and 0.2 micrometer roughness cases. For the 0.1 micrometer roughness case, this decrease happens sooner- at 0.001 kg/s.

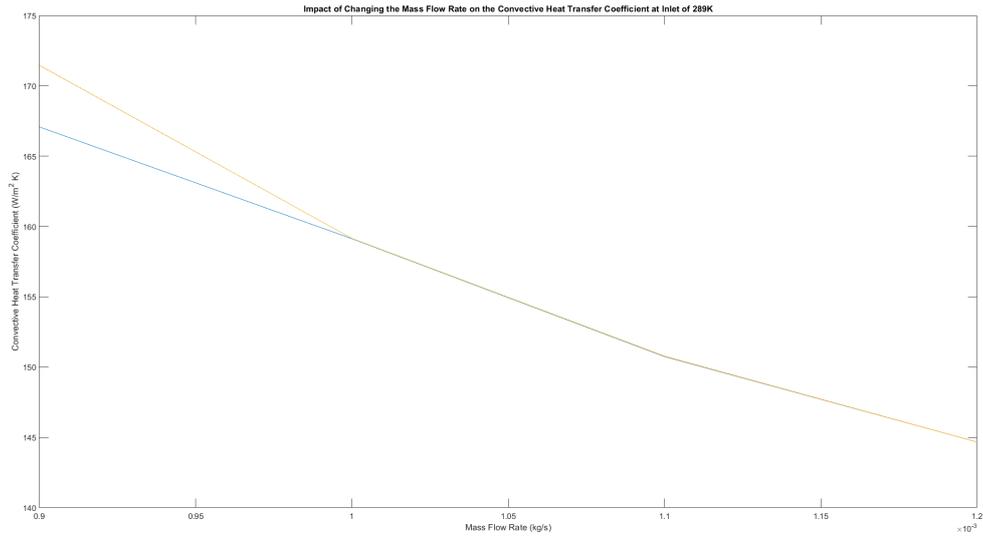


Figure 6.1: Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 289K

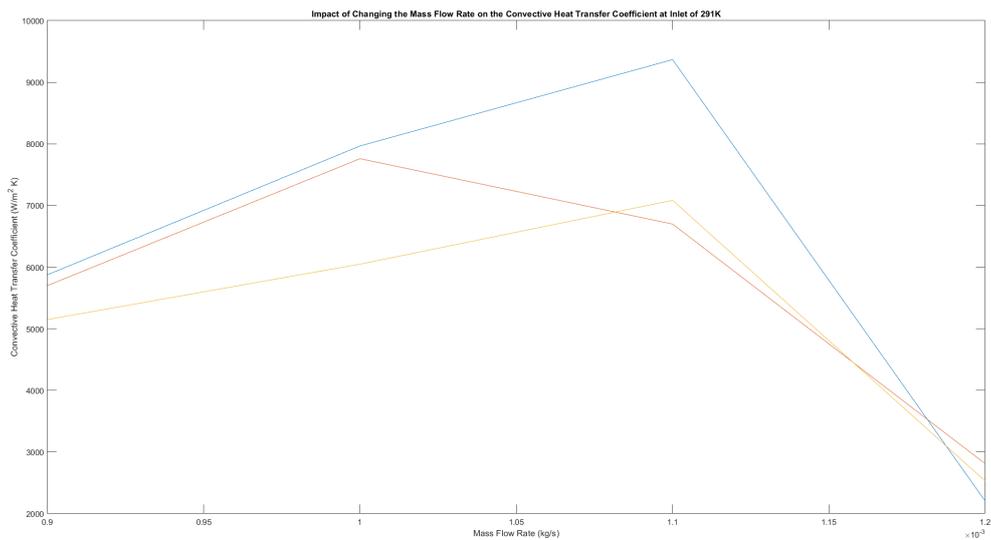


Figure 6.2: Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 291K

6.3 Effects of Varying Surface Roughness

The effect of changing surface roughness seems to be scattered. In the low temperature inlet case, Figure 6.7, increasing surface roughness increases the convective heat transfer coefficient. In the inlet at 291K (Figure 6.8), for some cases the heat transfer coefficient decreases with an increase in surface roughness, whereas in another, the convective heat transfer coefficient increases slightly.

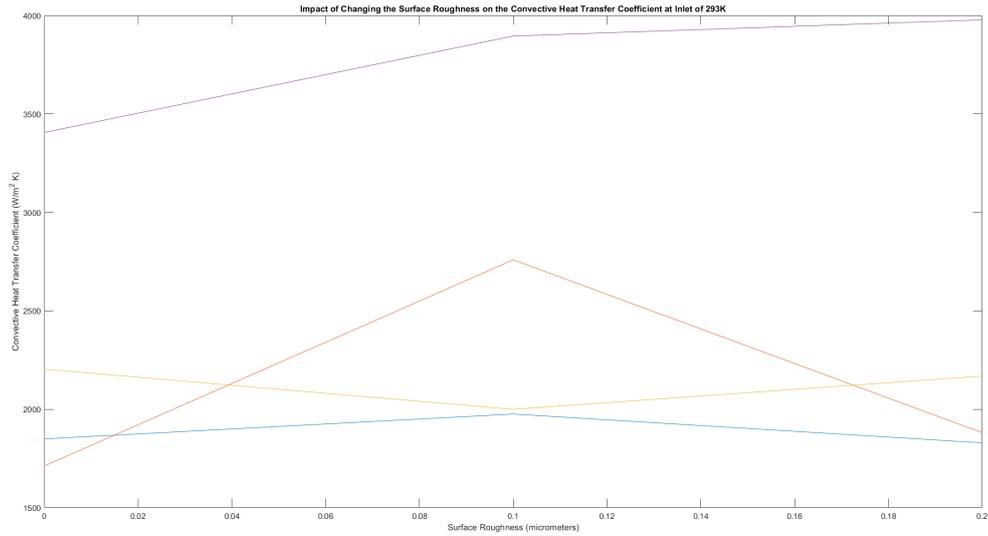


Figure 6.3: Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 293K

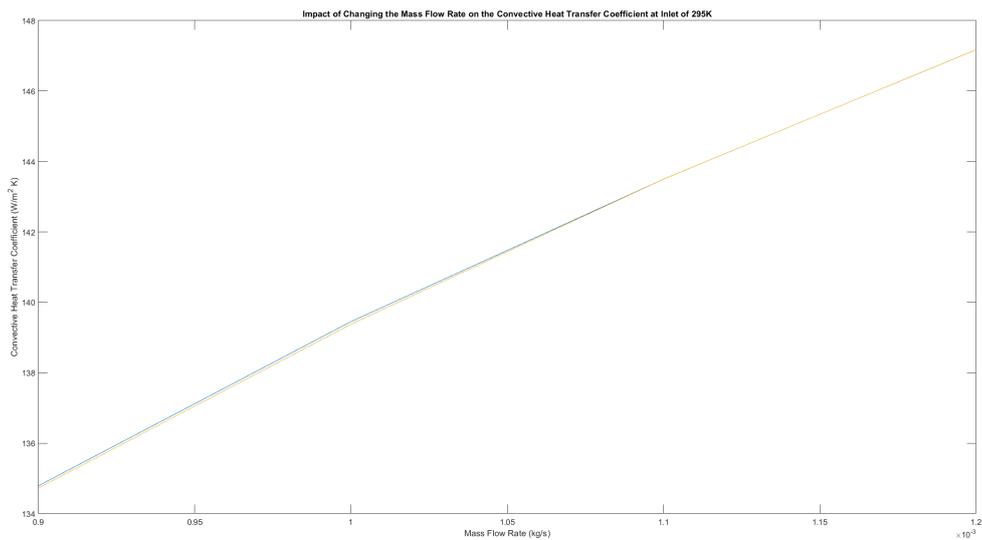


Figure 6.4: Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 295K

6.4 Effects of Varying Flow Inlet Temperature

In Figure 6.13 increasing the inlet temperature has a large increasing effect on the convective heat transfer coefficient. In the Nusselt number correlation calculations, this increase in inlet temperature would make a minor impact on the resulting heat transfer coefficient as a result of the fluid properties changing slightly.

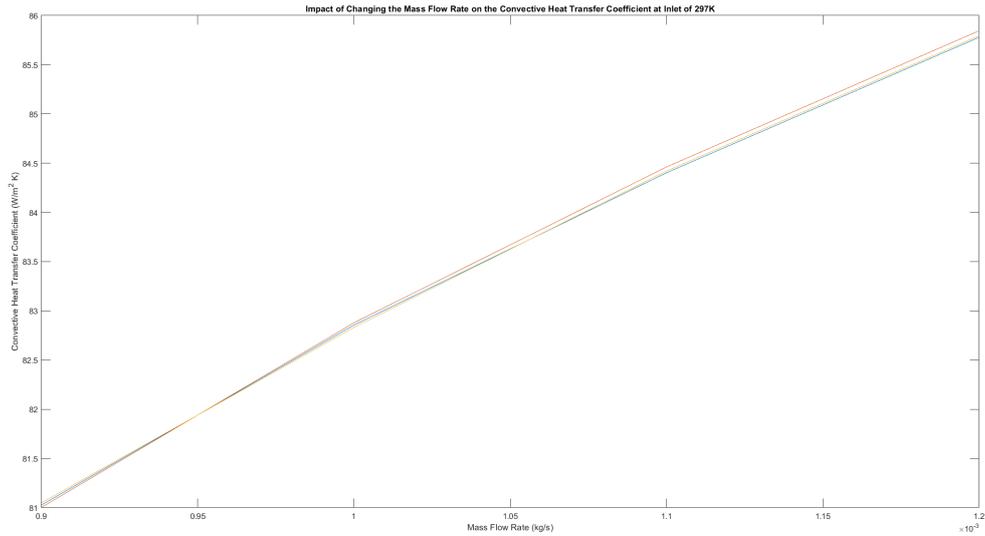


Figure 6.5: Effect of changing mass flow rate on convective heat transfer coefficient with inlet temperature of 297K

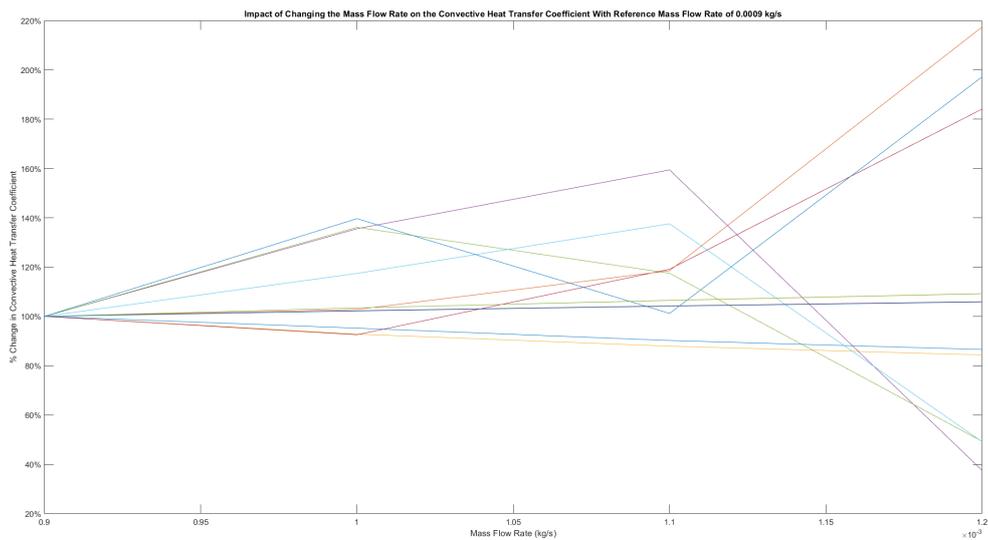


Figure 6.6: Effect of changing mass flow rate on convective heat transfer coefficient as compared to 0.0009 kg/s reference case

6.5 Surface Temperature vs Bulk Outlet Temperature

As Shown in Figure 6.14, the temperature at the fluid interface and the bulk average fluid temperature are very similar.

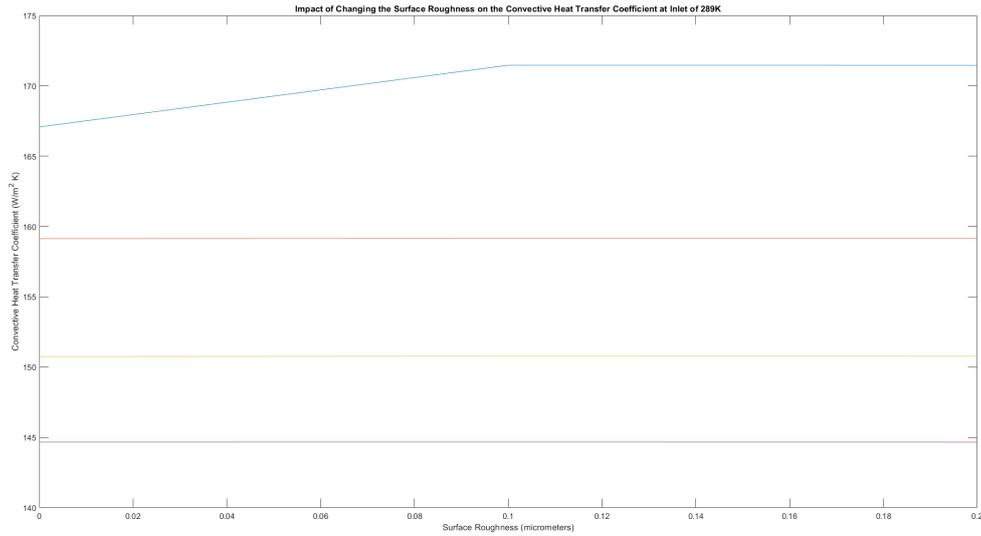


Figure 6.7: Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 289K

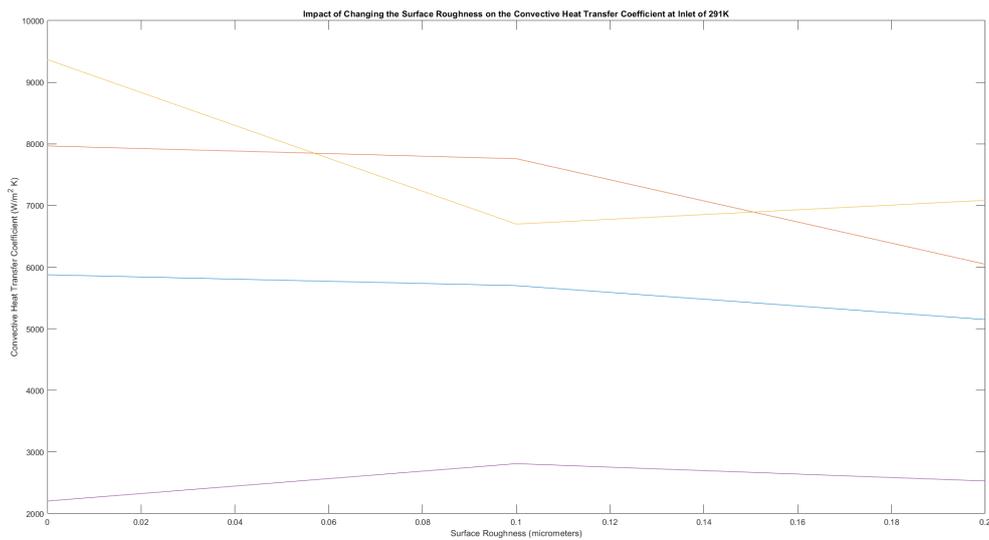


Figure 6.8: Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 291K

6.6 Comparison of CFD Heat Transfer Coefficient With Nusselt Number Calculations

For the first 12 cases in Table 6.3, the turbulent flow appears to be the best correlated with the values found from CFD. These cases are the 289K inlet flow cases, and none of the calculated convective heat transfer coefficients seem to be a great fit. For the 291K cases (13-24), the laminar Nusselt correlation appears to approximate most of the cases fairly well. The rest of the cases are shown in Appendix B.

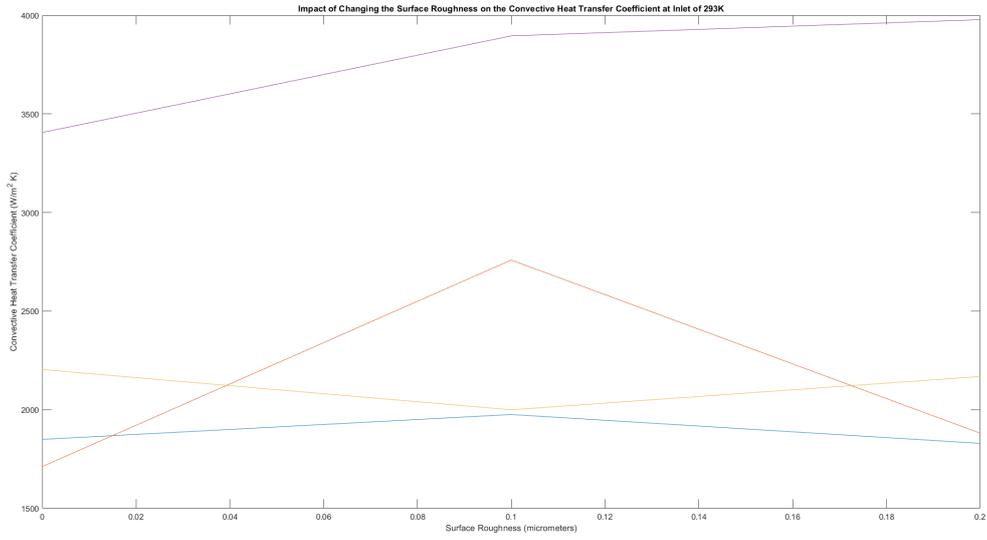


Figure 6.9: Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 293K

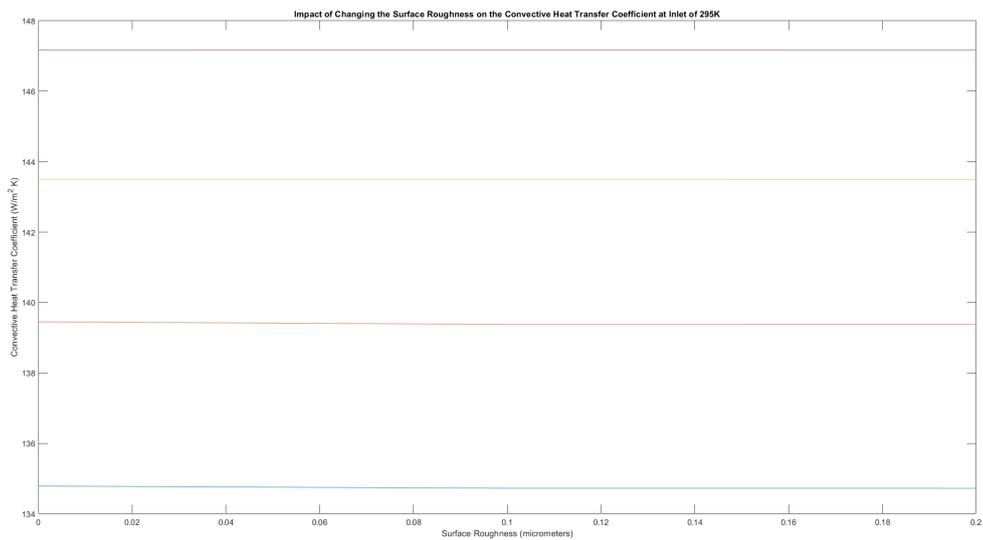


Figure 6.10: Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 295K

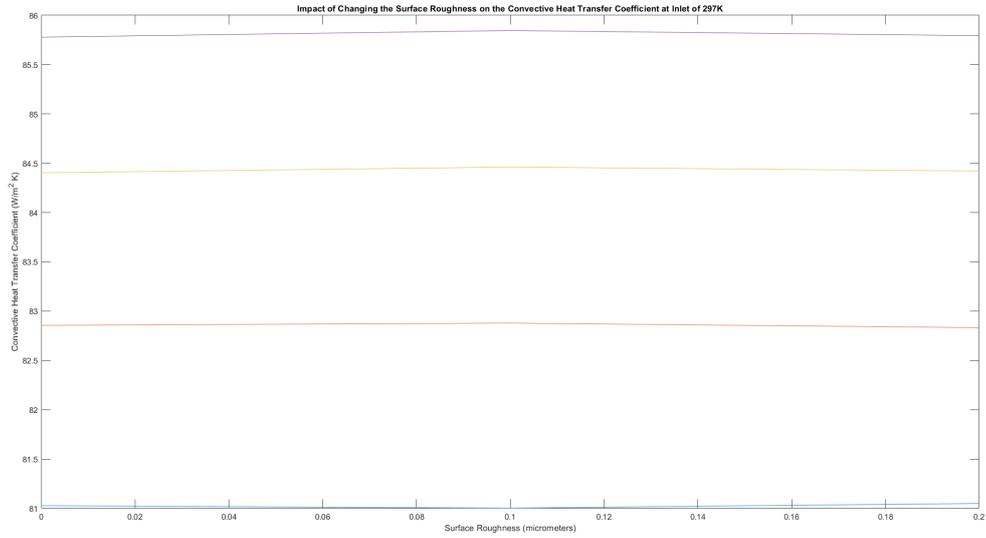


Figure 6.11: Effect of changing surface roughness on convective heat transfer coefficient with inlet temperature of 297K

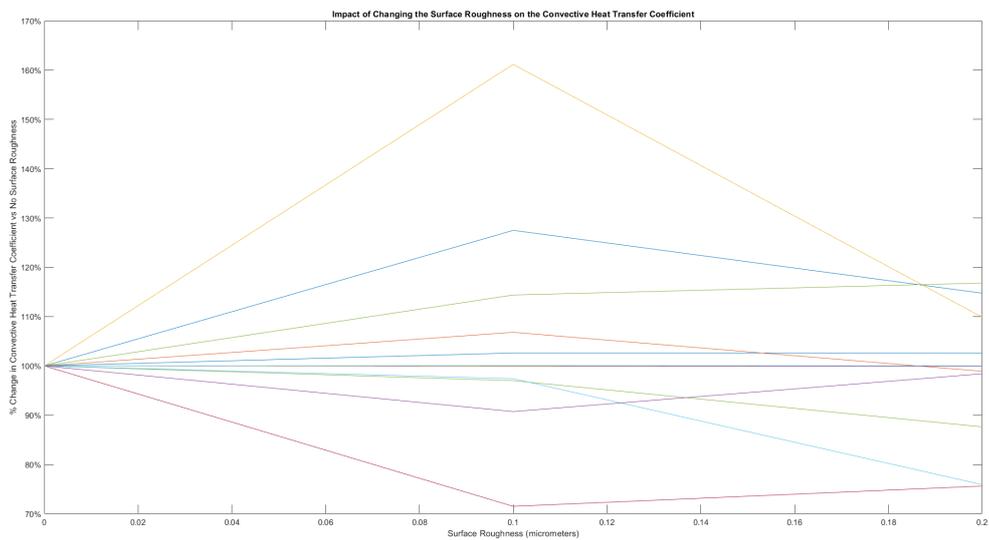


Figure 6.12: Effect of changing surface roughness on convective heat transfer coefficient as compared to zero roughness case

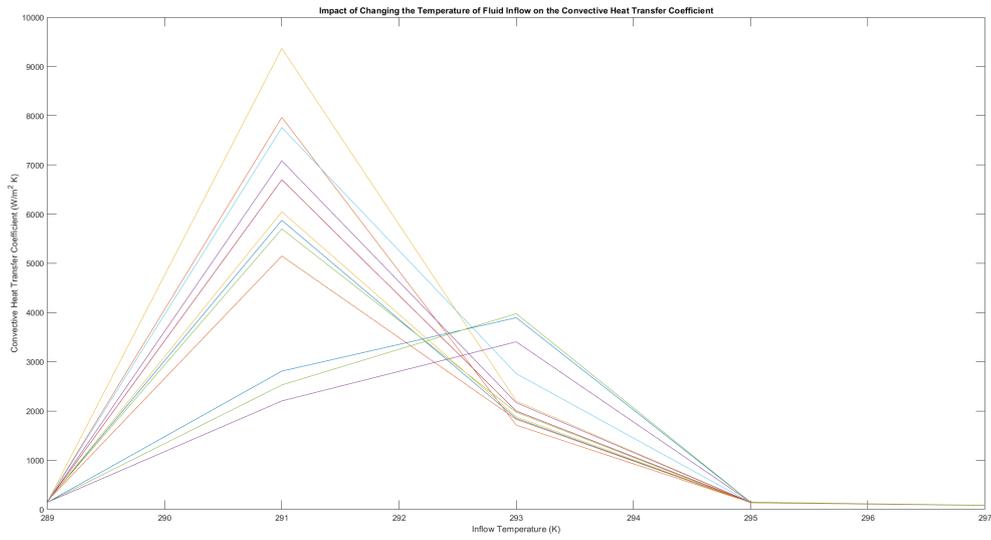


Figure 6.13: Effect of changing fluid inlet temperature on convective heat transfer coefficient

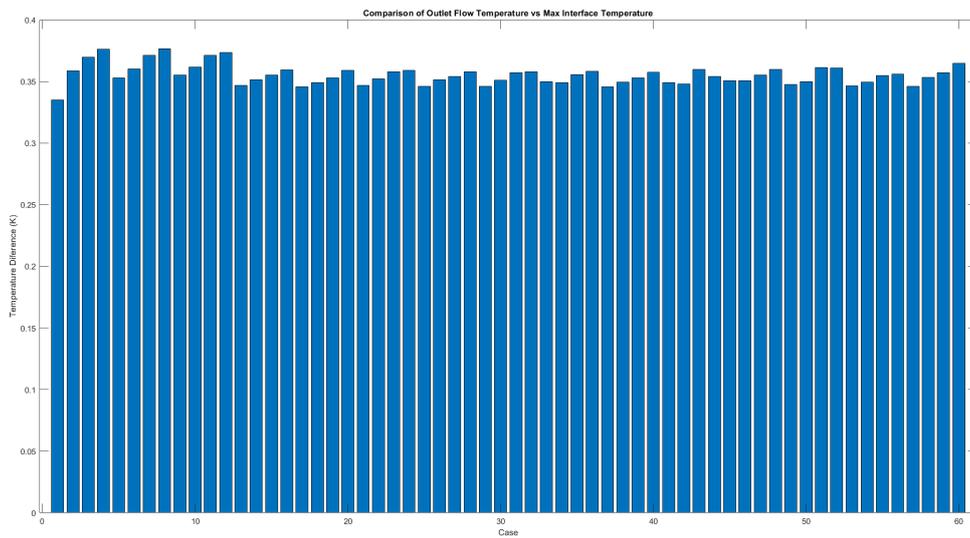


Figure 6.14: Difference between maximum interface temperature and bulk outflow temperature

Table 6.3: Heat Transfer Coefficient Comparison, Values in $\frac{W}{m^2K}$

Case	\bar{h}_{CFD}	$\bar{h}_{Laminar}$	$\bar{h}_{turbulent}$	$\bar{h}_{constflux}$
1	167.08	5454.18	513.85	6458.80
2	159.13	5459.28	587.35	6458.80
3	150.74	5462.50	633.89	6458.80
4	144.67	5465.66	679.59	6458.80
5	171.46	5455.98	539.87	6458.80
6	159.15	5459.28	587.35	6458.80
7	150.77	5462.50	633.89	6458.80
8	144.67	5465.66	679.59	6458.80
9	171.46	5455.98	539.87	6458.80
10	159.14	5459.28	587.35	6458.80
11	150.77	5462.50	633.89	6458.80
12	144.66	5465.66	679.59	6458.80
13	5875.29	5456.00	540.07	6458.80
14	7965.61	5459.29	587.57	6458.80
15	9366.99	5462.52	634.12	6458.80
16	2204.11	5465.68	679.83	6458.80
17	5699.71	5456.00	540.07	6458.80
18	7759.44	5459.29	587.57	6458.80
19	6698.56	5462.52	634.12	6458.80
20	2810.21	5465.68	679.83	6458.80
21	5149.17	5456.00	540.07	6458.80
22	6047.68	5459.29	587.57	6458.80
23	7083.71	5462.52	634.12	6458.80
24	2529.28	5465.68	679.83	6458.80

Chapter 7: Discussion and Analysis

The results from the cases ran indicate that the 1D heat transfer correlations do not match well with the CFD results. Comparison of the laminar Nusselt number correlations show an over-prediction by as much as 13x over the CFD calculated results. Clearly, in the cases with no surface roughness laminar flow is a good assumption to be made as no cases have a Reynolds number above 20, so an accounting of potential reasons behind the discrepancy between the Nusselt correlations and the CFD results is necessary to determine the validity of the CFD results and to guidance for future work exploring convective heat transfer in on-chip cooling.

The first key point to evaluate is how the properties of water compare between the CFD results and the one-dimensional calculations used. For the correlation comparison, the Prandtl number was considered constant. In each of the cases ran, the temperature range between inlet flow and outlet flow was less than 3 Kelvin. A property table of the Prandtl number shows a variation of 35.3% from the range of 280K to 290K [45], or if linearly interpolated, 10.6% in the 3 Kelvin range of fluid temperatures found in the CFD solution. While this can amount to a significant source of error, on its own it does not account for the discrepancy between the analytic calculation and the CFD calculation.

Another potential source of error comes from the assumptions of either constant surface temperature or constant heat flux through the surface. The results of the CFD show that these assumptions are either fundamentally flawed for this problem or that the correlations fit to the laboratory data do not work at this scale. One other Nusselt number correlation was found that was derived from experimental data. This equation is

$$Nu_w = 0.339Re^{0.6}Pr^{3.9} \quad (7.1)$$

and was found by curve fitting for carbon dioxide, nitrogen, and helium in microfluidics flow [46]. For the test case in the sample calculations, this gives a value of

$$Nu = 3137.50$$
$$\bar{h} = 4647815 \frac{W}{m^2K}$$

for an error of 2781690% as compared to the CFD results.

Because of these large discrepancies between curve fit Nusselt number correlations and CFD results, a new methodology of approximating heat transfer in microfluidics cases is proposed. The CFD solutions show that the bulk outlet temperature of the fluid is at most 0.13% lower than the maximum surface temperature of the interface between the fluid and the EVA layer. A potential way of solving these one-dimensional models where the convective heat transfer coefficient is inaccurate is to take the formula

$$q = \dot{m}C_p(T_{m,1} - T_{m,0}) \quad (7.2)$$

and set the peak surface temperature equal to $T_{m,1}$. From there, a slab model of conduction through the rest of the model can be used to estimate maximum thermal loads throughout the chip. It is worth noting that this is a direct result of

$$\lim_{h \rightarrow \infty} \frac{q''}{h} = (T_s - T_m) \quad (7.3)$$

This means that the overestimation of the heat transfer coefficient due to the Nusselt number correlations are a better predictor than this methodology. Despite this, the maximum surface temperature being approximated as the maximum outlet temperature still leads to accurate results for maximum temperatures in the silicon. This method allows for quicker calculations than initially calculating the convective heat transfer coefficient and increases the design speed of these channels.

7.1 Proposed Model Predicted Average Silicon Maximum Temperatures vs CFD Results

The accounting for temperatures throughout the model is performed via the one-dimensional thermal resistance model shown in Figure 7.1.

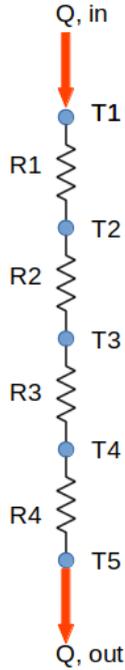


Figure 7.1: Solar panel thermal resistance model

Where Q_{in} is the heat generated on the top surface, T_1 is the top surface temperature of the glass, T_2 is the interface temperature between the glass and the top surface of the EVA top layer, T_3 is the interface temperature between the bottom face of the top EVA module and the top face of the Silicon, T_4 is the interface temperature between the bottom face of the silicon and the top face of the EVA bottom layer, and T_5 is the peak

temperature of the surface between the water flow and the EVA. The thermal resistances are based on the conduction model for thermal resistance and the material properties of each layer, where R_1 is the thermal resistance of the glass, R_2 is the thermal resistance of the top EVA layer, R_3 is the thermal resistance of the silicon, and R_4 is the thermal resistance of the bottom EVA layer.

From the thermal properties shown earlier,

$$R_1 = \frac{0.003(m)}{1.05(W/mK) * 0.01(m^2)} = 0.2857 \frac{W}{K}$$

$$R_2 = \frac{0.00045(m)}{0.34(W/mK) * 0.01(m^2)} = 0.1323 \frac{W}{K}$$

$$R_3 = \frac{0.00018(m)}{150(W/mK) * 0.01(m^2)} = 0.00012 \frac{W}{K}$$

$$R_4 = \frac{0.5 * (0.0006)(m)}{0.34(W/mK) * 0.01(m^2)} = 0.08824 \frac{W}{K}$$

The resistance of the final EVA layer is estimated by using half of the total thickness, as the convective flow is centered through the layer. The total heat transfer rate through the system is found by:

$$q = q'' A = 1000 \frac{W}{m^2} * 0.01m^2 = 10W$$

The temperature of the interface between the solar cell and bottom EVA can be found via

$$q = -\frac{T_5 - T_4}{R_4} = 10 = -\frac{291.65K - T_4}{0.08824}$$

$$T_4 = 292.53(K)$$

The top surface of the silicon's temperature can be found via

$$q = -\frac{T_4 - T_3}{R_3} = 10 = -\frac{292.53K - T_3}{0.00012}$$

$$T_3 = 292.53(K)$$

The CFD results have a maximum temperature in the silicon of 292.27 K, so for this example, using the maximum fluid temperature as the surface temperature over predicts by 0.26 K, or 0.09%. A graph of comparison evaluations between maximum solar cell temperature and this methodology for all CFD cases is shown in Figure 7.2.

This methodology has high levels of accuracy in the design chosen for this project. The maximum error across the sixty points ran is 0.09%, meaning that this design methodology has potential usefulness in the preliminary design for convective cooling designs of

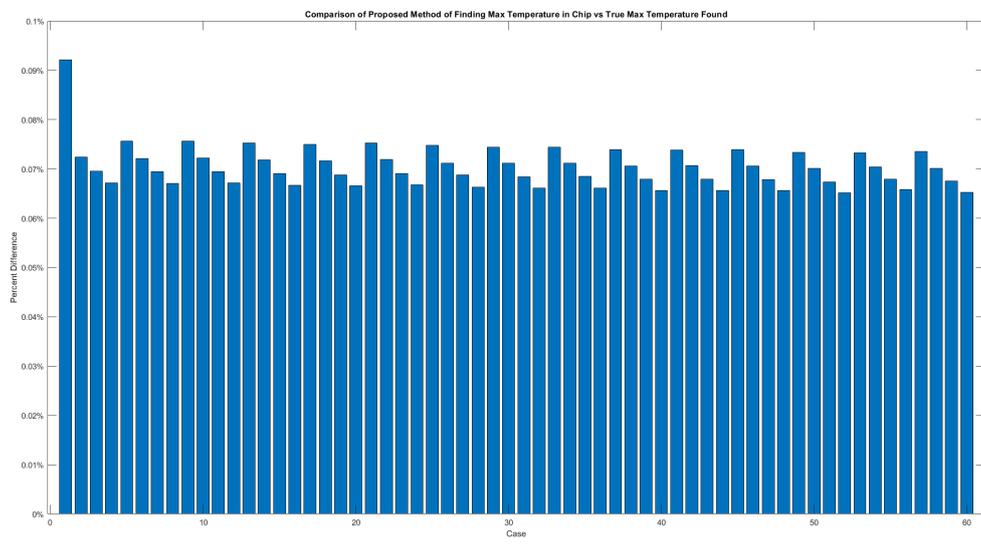


Figure 7.2: Error in maximum temperature prediction by proposed model

Chapter 8: Conclusions and Recommendations for Further Research

With an increase in high power technology, cooling of computer chips and power generation devices becomes more and more necessary. On chip cooling is frequently done via convective cooling, but methods of estimating the conductive heat transfer coefficient do not appear to accurately predict this coefficient in the cases ran. Because of this, if on-chip microfluidics cooling is to be investigated further, new correlations must be made to predict the heat removal and aid the design of these channels. This paper is an attempt towards a first step at exploring single-phase heat transfer with water in the backing layer of a photovoltaic cell.

Further research can and should be done with both lab testing and further CFD analysis. Lab testing would be needed to verify the CFD study, as the verification attempts performed here showed a lack of correlation with expected textbook values. Further CFD research should be performed to expand upon the work done in this paper. The variables explored within this paper (mass flow, surface roughness, and fluid inlet temperature) did not have a large enough range to adequately apply these results across an entire regime of flow. An increase in point density will also help provide a better method of fitting the convective heat transfer coefficient and could aid in design by removing the assumption of fluid outflow temperature being equal to maximum interface temperature..

Additional studies could also vary other properties such as the diameter of the micro-channels, their shape, and their spacing from each other to provide a picture of how these influence the convective properties. Comparison of this correlation can also be performed with another fluid to determine the effect of fluid properties such as the Prandtl number.

References

- [1] Pfreundt, A., Beinert, A., Yucebas, D., Mesquita, L. V., Pitta-Bauermann, L., Romer, P., and Mittag, M., “Post-processing thickness variation of PV module materials and its impact on temperature, MECHANICAL STRESS AND POWER,” 2019, https://www.ise.fraunhofer.de/content/dam/ise/de/documents/publications/conference-paper/36-eupvsec-2019/Pfreundt_4CO43.pdf.
- [2] ANSYS, “Turbulent pipe flow — entrance length,” 2020, <https://courses.ansys.com/index.php/courses/real-internal-flows/lessons/simulation-examples-homework-and-quizzes-4/topic/turbulent-pipe-flow-entrance-length-homework/>.
- [3] Anonymous, “Thermal resistance circuits,” 2022, <https://web.mit.edu/16.unified/www/FALL/thermodynamics/notes/node118.html>.
- [4] Svarc, J., “Solar panel construction,” 2020, <https://www.cleanenergyreviews.info/blog/solar-panel-components-construction>.
- [5] Anonymous, “10% vinyl acetate EVA,” 2020, <https://www.makeitfrom.com/material-properties/10-Percent-Vinyl-Acetate-EVA>.
- [6] SMIL, V., “Photovoltaics in satellites,” 2018, <https://spectrum.ieee.org/photovoltaics-in-satellites>.
- [7] Spremo, S., Crocker, A., and Panontin, T., “Small spacecraft overview,” 2019, <https://ntrs.nasa.gov/api/citations/20190031730/downloads/20190031730.pdf>.
- [8] Anonymous, “Keeping space chips cool and reliable,” 2021, <https://webb.nasa.gov/content/about/innovations/cryocooler.html>.
- [9] Spremo, S., Crocker, A., and Panontin, T., “James Webb telescope cryocooler,” 2019, <https://ntrs.nasa.gov/api/citations/20190031730/downloads/20190031730.pdf>.
- [10] Kumacheva, E., “Microfluidics,” 2012, https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiEwbKIzOX7AhXeN0QIHTQOAxcQFnoECB4QAQ&url=https%3A%2F%2Fboulderschool.yale.edu%2Fsites%2Fdefault%2Ffiles%2Ffiles%2FLecture_Boulder.pdf&usq=AOvVaw3PIKh2QLQfMpQ80Ir2Bf8i.
- [11] vanDam, R. M., “Solvent-resistant elastomeric microfluidic devices and applications,” 2006.
- [12] Adhikari, P., “Computational analysis of mixing in microchannels,” 2013, https://etd.ohiolink.edu/apexprod/rws_etd/send_file/send?accession=yasu1370799440&disposition=inline.

- [13] Song, Cheng, and Zhao, editors, *Microfluidics: fundamentals, devices, and applications*, Wiley, Weinheim, Germany, 2018, pp. 19, 56.
- [14] Ahmed, F., Yoshida, Y., Wang, J., and et al., “Design and validation of microfluidic parameters of a microfluidic chip using fluid dynamics,” *AIP Advances*, Vol. 11, 2021, pp. 1–6.
- [15] Marić, T., Kotheb, D., and Bothe, D., “Unstructured un-split geometrical volume-of-fluid methods – a review,” *Journal of Computational Physics*, Vol. 420, 2020.
- [16] “User inputs for time-dependent problems,” 2009, <https://www.afs.enea.it/project/neptunius/docs/fluent/html/udf/node2.htm>.
- [17] “OpenFOAM user guide,” 2021, <http://foam.sourceforge.net/docs/Guides-a4/OpenFOAMUserGuide-A4.pdf>.
- [18] CHEN, L., “Finite volume methods,” 2015, <https://www.math.uci.edu/~chenlong/226/FVM.pdf>.
- [19] “The Navier-Stokes equations,” 2021, https://www.iit.edu/sites/default/files/2021-02/navier_stokes.pdf.
- [20] Petrilu, T. and Trif, D., editors, *Basics of fluid mechanics and introduction to computational fluid dynamics*, chap. s, Springer Science, Boston, 2005.
- [21] Peyet, R., editor, *Handbook of computational fluid mechanics*, Elsevier Science and Technology, 1996.
- [22] Powell, D., Bogosian, D., and Schwer, L., “Mesh sensitivity of blast wave propagation,” Vol. 15 of *LS-DYNA Users Conference*, LS-DYNA, Dearborn, MI, 2018.
- [23] Tannehill, J., Anderson, D., and Pletcher, R., editors, *Computational fluid mechanics And heat transfer*, chap. 3, Taylor and Francis, Washington DC, 1997.
- [24] Lauritzen, P. H., “A stability analysis of finite-volume advection schemes permitting long time steps,” *Monthly Weather Review*, Vol. 135, 2006.
- [25] Peterson, J., “Introduction to boundary value problems,” 2017, https://people.sc.fsu.edu/~jpeterson/bvp_notes.pdf.
- [26] Zhang, Z., Wang, X., and Yan, Y., “A review of the state-of-the-art in electronic cooling,” *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, 2021.
- [27] Kumar, A., Javed, A., and Dubey, S., “Material selection for microchannel heatsink: conjugate heat transfer simulation,” Vol. 346, 2017.
- [28] Christianson, “The phases of matter,” 2019, [https://chem.libretexts.org/Courses/Bellarmino_University/BU%3A_Chem_104_\(Christianson\)/Phase_1%3A_The_Phases_of_Matter/1%3A_Gases/1.1%3A_The_Phases_of_Matter](https://chem.libretexts.org/Courses/Bellarmino_University/BU%3A_Chem_104_(Christianson)/Phase_1%3A_The_Phases_of_Matter/1%3A_Gases/1.1%3A_The_Phases_of_Matter).
- [29] Anonymous, “Fluids - lecture 12 notes,” 2012, <http://web.mit.edu/16.unified/www/SPRING/fluids/Spring2008/LectureNotes/f12.pdf>.

- [30] Anonymous, “Heat transfer: conservation of energy,” 2018, <https://www.comsol.com/multiphysics/heat-transfer-conservation-of-energy>.
- [31] Mei, C., “Chapter 4. thermal effects in fluids,” 2007, http://web.mit.edu/1.63/www/Lec-notes/chap4_heat_mass/4-1energy.pdf.
- [32] Thess, A., editor, *The finite volume method in computational fluid dynamics*, Springer, 2016.
- [33] Anonymous, “Sources of error in CFD,” 2018, <https://projects.exeter.ac.uk/fluidflow/ComputationalFluidDynamics/notes4web/notes4se1.html>.
- [34] Jasak, H., “Error analysis and estimation for the finite volume method with applications to fluid flows.” 1996, <https://spiral.imperial.ac.uk/handle/10044/1/8335>.
- [35] Bui, T., “Explicit and implicit methods in solving differential equations,” 2010, https://opencommons.uconn.edu/cgi/viewcontent.cgi?article=1118&context=srhonors_theses.
- [36] Anonymous, “Implicit vs. explicit numerical methods,” 2022, <https://www.flow3d.com/resources/cfd-101/numerical-issues/implicit-versus-explicit-numerical-methods/>.
- [37] Theresa, “How to keep the Courant number below one,” 2020, <https://www.simscale.com/knowledge-base/what-is-a-courant-number/>.
- [38] Anonymous, “Tips and tricks in OpenFOAM,” 2022, <http://www.wolfdynamics.com/wiki/tipsandtricks.pdf>.
- [39] Anonymous, “Energy transfer,” 2022, <https://www3.uwsp.edu/cnr-ap/KEEP/nres633/Pages/Unit2/Section-B-Energy-Transfer.aspx>.
- [40] Bergman, T., Lavine, A., Incropera, F., and Dewitt, D., editors, *Fundamentals of heat and mass transfer*, chap. 3, Wiley, Hoboken NJ, 2011.
- [41] Bergman, T., Lavine, A., Incropera, F., and Dewitt, D., editors, *Fundamentals of heat and mass transfer*, chap. 8, Wiley, Hoboken NJ, 2011.
- [42] Anonymous, “Water - dynamic (absolute) and kinematic viscosity vs. temperature and pressure,” 2004, https://www.engineeringtoolbox.com/water-dynamic-kinematic-viscosity-d_596.html.
- [43] Rapp, B. E., editor, *Microfluidics: modelling, mechanics and mathematics*, chap. 9, Elsevier, Amsterdam, 2016.
- [44] Anonymous, “Water - thermal conductivity vs. temperature,” 2018, https://www.engineeringtoolbox.com/water-liquid-gas-thermal-conductivity-temperature-pressure-d_2012.html.
- [45] Anonymous, “Water - prandtl number vs. temperature and pressure,” 2018, https://www.engineeringtoolbox.com/water-steam-Prandtl-number-d_2059.html.
- [46] Damean, N., Regtien, P. P., and Elwenspoek, M., “Heat transfer in a MEMS for microfluidics,” *Sensors and Actuators*, Vol. 105, 2003, pp. 137–149.

- [47] Capasso, V. and Bakstein, D., editors, *An introduction to continuous-time stochastic processes theory, models, and applications to finance, Biology, and Medicine*, Modeling and Simulation in Science, Engineering and Technology, Birkhauser Boston, Boston, 2005, pp. 51, 60.
- [48] Sobachkin, A. and Dumnov, G., “Numerical basis of CAD-embedded CFD,” 2014, https://www.solidworks.com/sw/docs/flow_basis_of_cad_embedded_cfd_whitepaper.pdf.
- [49] Anonymous, “Solar radiation,” 2020, <https://www.whoi.edu/science/AOPE/mvco/description/SolRad.html#:~:text=Above%20the%20earth%27s%20atmosphere%2C%20solar,noon%20in%20the%20summer%20months.>

Appendix A: Matlab Code

```
%Processing Max Temperature Data Solar Cell
close all
clear all
clc

%Constants:
k_water = 0.59255; %W/mK
k_glass = 1.05; %W/mK
k_EVA = 0.34; %W/mK
k_Silicon = 150; %W/mK

A = 0.01; %m^2
t_glass = 0.003; %m
t_EVA_top = 0.00045; %m
t_silicon = 0.00018; %m
t_EVA_bot = 0.0003; %m

HeatGen = 1000; %W/m^2

Pr_water = 6.9;
diam = 4*10^(-4); %m
nu_water = 1.0756*10^(-6); %m^2/s
L = 0.1; %m
HeatExchangeSurfaceArea = diam*pi*L*200;

T = readtable('SixtyPointsFinal.xlsx', 'ReadRowNames', true);

SurfaceRoughness = table2array(T(1,:));
MassFlow = table2array(T(6,:));
MaxInterfaceTemps = table2array(T(19,:));
MinInterfaceTemps = table2array(T(17,:));
FluidInFlowTemp = table2array(T(7,:));
FluidOutFlowTemp = table2array(T(13,:));
averageVelocity = (table2array(T(10,:)) + table2array(T(10,:))
    )/2;
MaxSiliconTemps = table2array(T(16,:));
CFDConvectiveHeatTransferCoeff = table2array(T(21,:));

Re_D = averageVelocity * diam / nu_water;
```

```

%Laminar Calculation
Gzd = diam/L * Re_D * Pr_water;
Nu_D_laminar = 3.66+(0.0668*Gzd)./(1+0.4*Gzd.^(2/3));
h_bar_laminar = Nu_D_laminar*k_water/diam;

%Turbulent Calculation
Nu_D_turbulent = 0.023*Re_D.^(4/5)*Pr_water^(0.3);
h_bar_turbulent = Nu_D_turbulent*k_water/diam;

%Constant Heat Flux Calculation
Nu_D_flux = 4.36;
h_bar_flux = Nu_D_flux*k_water/diam;

%Thermal Resistances:
R1 = t_glass/(k_glass*A);
R2 = t_EVA_top/(k_EVA*A);
R3 = t_silicon/(k_Silicon*A);
R4 = t_EVA_bot/(k_EVA*A);

q = HeatGen*A; %W

%Proposed Prediction Method
T5 = FluidOutFlowTemp;
T4 = R4*q+T5;
T3 = R3*q+T4;

deltaMaxTemps = T3 - MaxSiliconTemps;
percentDifferenceMaxPrediction = deltaMaxTemps ./
    MaxSiliconTemps * 100;

bar(percentDifferenceMaxPrediction)
ylabel('Percent Difference')
xlabel('Case')
title('Comparison of Proposed Method of Finding Max
    Temperature in Chip vs True Max Temperature Found')
ytickformat('percentage')

%%Surface Temp vs Bulk Outlet Temperature
figure
deltaT = MaxInterfaceTemps-FluidOutFlowTemp;
bar(deltaT)
ylabel('Temperature Diference (K)')
xlabel('Case')
title('Comparison of Outlet Flow Temperature vs Max Interface

```

```

    Temperature')

%%Effects of Varying Flow Inlet Temperature:
[h_inletTempAffects, inflowTemps] = seperateArray(12,
    CFDConvectiveHeatTransferCoeff, FluidInFlowTemp);
h_inletPercentAffects = h_inletTempAffects./
    h_inletTempAffects(1,:) * 100;

figure
plot(inflowTemps, h_inletTempAffects)
xlabel('Inflow Temperature (K)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Temperature of Fluid Inflow on
    the Convective Heat Transfer Coefficient')

%%Effects of Varying Surface Roughness:
[h_surfaceRoughEffects, surfRoughPlot] = seperateArray(4,
    CFDConvectiveHeatTransferCoeff, SurfaceRoughness);

h_surfaceRoughEffectsT1 = h_surfaceRoughEffects(1:3,:);
h_surfaceRoughEffectsT2 = h_surfaceRoughEffects(4:6,:);
h_surfaceRoughEffectsT3 = h_surfaceRoughEffects(7:9,:);
h_surfaceRoughEffectsT4 = h_surfaceRoughEffects(10:12,:);
h_surfaceRoughEffectsT5 = h_surfaceRoughEffects(13:15,:);
surfRoughPlot = surfRoughPlot(1:3,:);

figure
plot(surfRoughPlot, h_surfaceRoughEffectsT1)
xlabel('Surface Roughness (micrometers)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Surface Roughness on the
    Convective Heat Transfer Coefficient at Inlet of 289K')

figure
plot(surfRoughPlot, h_surfaceRoughEffectsT2)
xlabel('Surface Roughness (micrometers)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Surface Roughness on the
    Convective Heat Transfer Coefficient at Inlet of 291K')

figure
plot(surfRoughPlot, h_surfaceRoughEffectsT3)
xlabel('Surface Roughness (micrometers)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Surface Roughness on the
    Convective Heat Transfer Coefficient at Inlet of 293K')

```

```

figure
plot(surfRoughPlot, h_surfaceRoughEffectsT4)
xlabel('Surface Roughness (micrometers)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Surface Roughness on the
      Convective Heat Transfer Coefficient at Inlet of 295K')

```

```

figure
plot(surfRoughPlot, h_surfaceRoughEffectsT5)
xlabel('Surface Roughness (micrometers)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Surface Roughness on the
      Convective Heat Transfer Coefficient at Inlet of 297K')

```

```

percentChangeDueToSurfaceRoughnessT1 =
    h_surfaceRoughEffectsT1./(h_surfaceRoughEffectsT1(1,:))
    *100;
percentChangeDueToSurfaceRoughnessT2 =
    h_surfaceRoughEffectsT2./(h_surfaceRoughEffectsT2(1,:))
    *100;
percentChangeDueToSurfaceRoughnessT3 =
    h_surfaceRoughEffectsT3./(h_surfaceRoughEffectsT3(1,:))
    *100;
percentChangeDueToSurfaceRoughnessT4 =
    h_surfaceRoughEffectsT4./(h_surfaceRoughEffectsT4(1,:))
    *100;
percentChangeDueToSurfaceRoughnessT5 =
    h_surfaceRoughEffectsT5./(h_surfaceRoughEffectsT5(1,:))
    *100;

```

```

figure
plot(surfRoughPlot, percentChangeDueToSurfaceRoughnessT1)
hold on
plot(surfRoughPlot, percentChangeDueToSurfaceRoughnessT2)
plot(surfRoughPlot, percentChangeDueToSurfaceRoughnessT3)
plot(surfRoughPlot, percentChangeDueToSurfaceRoughnessT4)
plot(surfRoughPlot, percentChangeDueToSurfaceRoughnessT5)
xlabel('Surface Roughness (micrometers)')
ylabel('% Change in Convective Heat Transfer Coefficient vs
      No Surface Roughness')
title('Impact of Changing the Surface Roughness on the
      Convective Heat Transfer Coefficient')
ytickformat('percentage')

```

```

%Effect of Varying Mass Flow Rate on Convective Heat
Transfer Coefficient:

```

```

[h_massFlowEffects, massFlowRates] = seperateArray(4,
    CFDConvectiveHeatTransferCoeff, MassFlow);

```

```

h_massFlowPercentAffects = h_massFlowEffects ./
    h_massFlowEffects(1,:) * 100;

h_massFlowEffectsT1 = h_massFlowEffects(1:3,:);
h_massFlowEffectsT2 = h_massFlowEffects(4:6,:);
h_massFlowEffectsT3 = h_massFlowEffects(7:9,:);
h_massFlowEffectsT4 = h_massFlowEffects(10:12,:);
h_massFlowEffectsT5 = h_massFlowEffects(13:15,:);
massFlowRatesPlot = massFlowRates(1:3,:);

figure
plot(massFlowRatesPlot, h_massFlowEffectsT1)
xlabel('Mass Flow Rate (kg/s)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Mass Flow Rate on the
    Convective Heat Transfer Coefficient at Inlet of 289K')

figure
plot(massFlowRatesPlot, h_massFlowEffectsT2)
xlabel('Mass Flow Rate (kg/s)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Mass Flow Rate on the
    Convective Heat Transfer Coefficient at Inlet of 291K')

figure
plot(massFlowRatesPlot, h_massFlowEffectsT3)
xlabel('Mass Flow Rate (kg/s)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Mass Flow Rate on the
    Convective Heat Transfer Coefficient at Inlet of 293K')

figure
plot(massFlowRatesPlot, h_massFlowEffectsT4)
xlabel('Mass Flow Rate (kg/s)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Mass Flow Rate on the
    Convective Heat Transfer Coefficient at Inlet of 295K')

figure
plot(massFlowRatesPlot, h_massFlowEffectsT5)
xlabel('Mass Flow Rate (kg/s)')
ylabel('Convective Heat Transfer Coefficient (W/m^2 K)')
title('Impact of Changing the Mass Flow Rate on the
    Convective Heat Transfer Coefficient at Inlet of 297K')

percentChangeDueToMassFlowT1 = h_massFlowEffectsT1 ./ (
    h_massFlowEffectsT1(1,:)) * 100;

```

```

percentChangeDueToMassFlowT2 = h_massFlowEffectsT2./(
    h_massFlowEffectsT2(1,:))*100;
percentChangeDueToMassFlowT3 = h_massFlowEffectsT3./(
    h_massFlowEffectsT3(1,:))*100;
percentChangeDueToMassFlowT4 = h_massFlowEffectsT4./(
    h_massFlowEffectsT4(1,:))*100;
percentChangeDueToMassFlowT5 = h_massFlowEffectsT5./(
    h_massFlowEffectsT5(1,:))*100;
figure
plot(massFlowRatesPlot, percentChangeDueToMassFlowT1)
hold on
plot(massFlowRatesPlot, percentChangeDueToMassFlowT2)
plot(massFlowRatesPlot, percentChangeDueToMassFlowT3)
plot(massFlowRatesPlot, percentChangeDueToMassFlowT4)
plot(massFlowRatesPlot, percentChangeDueToMassFlowT5)
xlabel('Mass Flow Rate (kg/s)')
ylabel('% Change in Convective Heat Transfer Coefficient')
title('Impact of Changing the Mass Flow Rate on the
    Convective Heat Transfer Coefficient With Reference Mass
    Flow Rate of 0.0009 kg/s')
ytickformat('percentage')

h_table = [[1:60]', CFDConvectiveHeatTransferCoeff',
    h_bar_laminar', h_bar_turbulent', h_bar_flux*ones(60,1)];
sympref('FloatingPointOutput',1)
latex_table = latex(sym(h_table))

function [seperatedArray1, seperatedArray2] = seperateArray(n
    , Array1, Array2)
if length(Array1) == length(Array2) && mod(length(Array2),n)
    ==0
    for i = 1:length(Array2)/n
        %If n = 12, then we want: 1:12, 13:24, 25: 36, n*i-n
        +1: n*i
        seperatedArray1(i,:) = Array1(n*i-n+1: n*i);
        seperatedArray2(i,:) = Array2(n*i-n+1: n*i);
    end
else
    seperatedArray1 = 0;
    seperatedArray2 = 0;
end
end

```

Appendix B: Full Table of Heat Transfer Coefficient Results

Table B.1: Heat Transfer Coefficient Comparison, Values in $\frac{W}{m^2K}$

Case	h_{CFD}	$h_{Laminar}$	$h_{turbulent}$	$h_{constflux}$
1	167.0778	5.4542e+03	513.8503	6.4588e+03
2	159.1285	5.4593e+03	587.3526	6.4588e+03
3	150.7443	5.4625e+03	633.8887	6.4588e+03
4	144.6673	5.4657e+03	679.5851	6.4588e+03
5	171.4643	5.4560e+03	539.8746	6.4588e+03
6	159.1475	5.4593e+03	587.3526	6.4588e+03
7	150.7727	5.4625e+03	633.8887	6.4588e+03
8	144.6714	5.4657e+03	679.5851	6.4588e+03
9	171.4599	5.4560e+03	539.8746	6.4588e+03
10	159.1478	5.4593e+03	587.3526	6.4588e+03
11	150.7654	5.4625e+03	633.8887	6.4588e+03
12	144.6644	5.4657e+03	679.5851	6.4588e+03
13	5.8753e+03	5.4560e+03	540.0714	6.4588e+03
14	7.9656e+03	5.4593e+03	587.5667	6.4588e+03
15	9.3670e+03	5.4625e+03	634.1198	6.4588e+03
16	2.2041e+03	5.4657e+03	679.8329	6.4588e+03
17	5.6997e+03	5.4560e+03	540.0714	6.4588e+03
18	7.7594e+03	5.4593e+03	587.5667	6.4588e+03
19	6.6986e+03	5.4625e+03	634.1198	6.4588e+03
20	2.8102e+03	5.4657e+03	679.8329	6.4588e+03
21	5.1492e+03	5.4560e+03	540.0714	6.4588e+03
22	6.0477e+03	5.4593e+03	587.5667	6.4588e+03
23	7.0837e+03	5.4625e+03	634.1198	6.4588e+03
24	2.5293e+03	5.4657e+03	679.8329	6.4588e+03
25	1.8496e+03	5.4560e+03	540.2783	6.4588e+03
26	1.7120e+03	5.4593e+03	587.7918	6.4588e+03
27	2.2037e+03	5.4625e+03	634.3628	6.4588e+03
28	3.4059e+03	5.4657e+03	680.0934	6.4588e+03
29	1.9757e+03	5.4560e+03	540.2783	6.4588e+03
30	2.7593e+03	5.4593e+03	587.7918	6.4588e+03
31	1.9997e+03	5.4625e+03	634.3628	6.4588e+03
32	3.8958e+03	5.4657e+03	680.0934	6.4588e+03
33	1.8295e+03	5.4560e+03	540.2783	6.4588e+03
34	1.8811e+03	5.4593e+03	587.7918	6.4588e+03
35	2.1686e+03	5.4625e+03	634.3628	6.4588e+03
36	3.9781e+03	5.4657e+03	680.0934	6.4588e+03
37	134.7881	5.4560e+03	540.4982	6.4588e+03
38	139.4523	5.4593e+03	588.0310	6.4588e+03
39	143.4984	5.4626e+03	634.6209	6.4588e+03
40	147.1775	5.4657e+03	680.3701	6.4588e+03
41	134.7238	5.4560e+03	540.4982	6.4588e+03
42	139.3788	5.4593e+03	588.0310	6.4588e+03
43	143.4994	5.4626e+03	634.6209	6.4588e+03
44	147.1751	5.4657e+03	680.3701	6.4588e+03
45	134.7231	5.4560e+03	540.4982	6.4588e+03
46	139.3797	5.4593e+03	588.0310	6.4588e+03
47	143.4968	5.4626e+03	634.6209	6.4588e+03
48	147.1779	5.4657e+03	680.3701	6.4588e+03
49	81.0268	5.4560e+03	540.7284	6.4588e+03
50	82.8557	5.4593e+03	588.2815	6.4588e+03
51	84.4027	5.4626e+03	634.8912	6.4588e+03
52	85.7787	5.4657e+03	680.6599	6.4588e+03
53	81.0027	5.4560e+03	540.7284	6.4588e+03
54	82.8789	5.4593e+03	588.2815	6.4588e+03
55	84.4619	5.4626e+03	634.8912	6.4588e+03
56	85.8456	5.4657e+03	680.6599	6.4588e+03
57	81.0494	5.4560e+03	540.7284	6.4588e+03
58	82.8321	5.4593e+03	588.2815	6.4588e+03
59	84.4208	5.4626e+03	634.8912	6.4588e+03
60	85.7938	5.4657e+03	680.6599	6.4588e+03