

A Cyclic Heat Transfer Solver for OpenFOAM

Michael Coe¹, * and Daniel Holland¹

¹University of Canterbury, Private Bag 4800, Christchurch 8140, New Zealand
Email address: michael.coe@canterbury.ac.nz

DOI: <https://doi.org/10.51560/ofj.v3.113>

Results with version(s): OpenFOAM® v2206

Repository: <https://github.com/michaelcoe/cyclicHeatTransfer-OpenFOAM>

Abstract. Channels with periodically repeating geometries are often simulated using periodic or cyclic boundary conditions. By calculating the temperature and flow field in one periodic module, the resulting distributions can be generalized to multiple modules. This reduces the computational load by simulating a single module versus the whole structure. This is a particularly useful approach when performing large optimisation studies of periodic geometries, such as compact heat exchangers. Currently, OpenFOAM only supports cyclic boundary conditions for pressure and momentum, but not heat transfer. The present work introduces a steady and an unsteady solver for cyclic heat transfer with constant wall temperature boundary conditions. The solver is validated against analytical Hagen-Poiseuille flow and two configurations of periodic wavy channels. In the latter case, the results are compared to existing literature.

1. Introduction

Heat conduction and diffusion of fluid in ducts of various cross-sectional shapes has been studied for many years. A number of technically important heat exchanger and duct configurations possess cross sections of flow that vary periodically [1]. An example of this is an offset plate-fin heat exchanger in which the channel bounding walls are periodically interrupted by gaps. Another example of periodic structures is wavy or corrugated channels, meant to augment heat transfer [2,3]. In these situations, the computational demand required to simulate the entire domain is likely prohibitive. As an alternative approach, the domain is taken such that the structure is periodic, and therefore the heat transfer can be characterised by simulating a much smaller volume using periodic boundary conditions. This paper implements a periodic boundary condition for heat transfer with a constant wall temperature. An appropriate multi-scale procedure where the appropriate heat transfer metrics can be scaled from a unit cell, as in Ciuffini *et al.* [4], can be used to determine the performance of the larger structure. This reduction in computation time is particularly significant when performing large scale optimisation on a structure.

The first generalized description of appropriate temperature transforms and boundary treatments for periodic flow, in both pressure and temperature fields, was proposed by Patankar *et al.* [1]. This description was valid for fully developed flow with two dimensional or axisymmetric duct flow in the laminar regime. Patankar *et al.* considered two heat transfer cases: constant wall temperature (CWT) and constant surface heat flux (SHF). For the CWT case, the temperature decay rate, λ_L is an unknown function that is approximated and the temperature field is normalized such that the source terms for the energy equation are only a function of a decay rate. For the SHF case, the source terms for the energy equation are only a function of the velocity and a temperature difference with no need to solve for an unknown $\lambda(x)$.

To make the calculation of $\lambda(x)$ easier, Stalio and Piller [5] introduced a normalized temperature based on the temperature transform from Patankar *et al.* [1]. This new transform allowed the decay rate to be the root of a quadratic equation based on the integration of the energy equation over the simulation domain. This model was extended by Wang *et al.* [6] so that the decay rate became a simple energy balance between the flux entering and leaving the periodic cell. The advantage of this method is that the decay rate could be calculated without the volumetric integration of the entire simulation domain. Currently,

* Corresponding author

the Patankar *et al.* model is available in commercial codes such as Ansys[®] Fluent [7], but only periodic (cyclic) flow for pressure and momentum is supported by OpenFOAM. There exists other boundary condition options such as **jumpCyclic** [8], where temperature value between patches can be specified by a uniform or non-uniform jump. The temperature field is then calculated based on the difference between the inlet and outlet temperatures. This condition is useful if the outlet temperature is known beforehand, but cannot be used if you specify a constant wall temperature and the final temperature field is unknown. The approach of Patankar *et al.* [1] only prescribes the shape of the temperature field and the field itself is calculated.

There have been studies involving periodic flow with SHF [9] boundary conditions utilizing solver code developed by Hærvig [10]. This code only supports SHF flows which involves an extra source term on the energy equation with no need to solve for the decay rate, $\lambda(x)$, that is needed for the CWT case. The CWT case is a valid model for many other heat transfer cases, so this work focuses on that model. A successful implementation of the Wang *et al.* [6] model, with CWT boundary conditions, exists for OpenFOAM, and was used to study 2D [11] wavy channels and later extended to 3D [12].

Where this work differs from the above works is in the implementation of a steady solver for cyclic heat transfer with CWT boundary conditions. The above works have only addressed unsteady flows of a fluid at Prandtl 0.7. This work implements a similar solver model for both steady (SIMPLE) and unsteady (PIMPLE) algorithms and seeks to give a comprehensive validation and characterization of these two solver. A further difference between this work and the above works is the inclusion of fluids with different Prandtl numbers. This work considers two fluids with Prandtl numbers 0.706 and 4.834, representing air and water, and there is a striking difference in solver behavior between the two fluids at the same Reynolds number.

This work is organized as follows: Section 2 presents the mathematical basis of decomposing the flow field into a periodic pressure and temperature field. Section 3 describes hows the solver is implemented to solve for the temperature field. Section 4 presents the solver validation with 2D Hagen-Poiseuille Flow and wavy channel geometries. This section further provides a comparison of 2D wavy channel simulations with published results.

2. Theoretical Background

Following the work of Patankar *et al.* [1] and the subsequent works of Stalio and Pillar [5] and Wang *et al.* [6], the pressure and temperature fields are decomposed into a global mass flow of the system and local fluctuations. Using the argument and formulation presented in [6], the energy flux through a single module is used to determine the temperature field. This section provides the mathematical basis for the cyclic heat transfer solver.

2.1. Theory of Periodic Mass Flow. For fully developed flows within a periodic module, the flow field is driven by a pressure gradient governed by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0 \quad (1)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = -\frac{1}{\rho} \Delta P + \nu \nabla^2 \mathbf{u}. \quad (2)$$

where ρ is the fluid density, P is the pressure, $\mathbf{u}$ is the velocity flow vector, ν is the kinematic viscosity, and t is the time. When the flow is periodic within a module, the velocity vector follows

$$\mathbf{u}(x \pm mL, y) = \mathbf{u}(x, y), \quad (3)$$

where L is the length of the periodic unit in the stream-wise direction, and m is an integer. The fluid pressure will decrease along the flow direction with the following relationship [1]:

$$P(x, y) = -\beta x + \bar{P}(x, y), \quad (4)$$

where β is a constant defined as

$$\beta = \frac{P(x, y) - P(x \pm mL, y)}{L}. \quad (5)$$

Here βx is related to the global mass flow of the system and $\bar{P}(x, y)$ represents the local pressure fluctuations.

In practice, running a cyclic simulation with OpenFOAM, the pressure gradient is calculated and constrained with the **meanVelocityForce** entry in the **fvOptions** or **fvConstraints** file, depending on the version. This option lets a user specify an average flow rate through a volume that gets recalculated

every timestep. The pressure gradient is then changed such that the volume averaged velocity is a user-defined function. A detailed description of the implementation of such a constraint with validation is provided by [13].

2.2. Theory of Periodic Heat Transfer. For 2D laminar flow, the temperature field is computed by the energy equation

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \alpha \nabla^2 T, \quad (6)$$

where $\mathbf{u}$ is the flow velocity, α is the thermal diffusivity, and T is the temperature. Here, we assume an incompressible fluid with a constant Prandtl number and a negligible viscous dissipation term [14]. For hydro-dynamically fully developed flow,

$$\frac{\partial^2 T}{\partial x^2} = 0, \quad (7)$$

which reduces the energy equation to

$$\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} = \alpha \left(\frac{\partial^2 T}{\partial y^2} \right). \quad (8)$$

For the CWT case, following the work of [6], the wall temperature is subtracted from the temperature field,

$$\theta = T(x, y) - T_w, \quad (9)$$

where θ is the reduced temperature, and T_w is the wall temperature. The wall boundary conditions ensure that $\theta(\Omega) = 0$, where Ω is the wall surface. The periodic relationship for the temperature field is assumed to follow the form:

$$\theta(x \pm mL, y) = e^{-\lambda_L(\pm mL)} \theta(x, y), \quad (10)$$

where λ_L is a decay constant in the streamwise direction [5]. Stalio and Piller [5] introduced a periodic temperature field, $\bar{\theta}$, divided by the exponential decay as

$$\bar{\theta}(x, y) = \frac{\theta(x, y)}{e^{-\lambda_L x}} = \frac{T(x, y) - T_w}{e^{-\lambda_L x}}. \quad (11)$$

Substituting Eqn. 11 into the energy equation, Eqn. 6 gives

$$\frac{\partial \bar{\theta}}{\partial t} + \mathbf{u} \cdot \nabla \bar{\theta} = \alpha \nabla^2 \bar{\theta} + (\alpha \lambda_L^2 + \lambda_L u_x) \bar{\theta} - 2\alpha \lambda_L \frac{\partial \bar{\theta}}{\partial x}, \quad (12)$$

where u_x is the velocity component in the flow direction and α is the thermal diffusivity of the fluid. Equation 12 satisfies the boundary condition of

$$\bar{\theta}(x \pm mL, y) = \bar{\theta}(x, y). \quad (13)$$

After solving for $\bar{\theta}$, the field is divided by the average of the inlet in order to keep the mean inlet $\bar{\theta}_{in,mean} = 1$, as done in [1, 6, 11, 12]. This is a necessary condition to prevent the temperature field from increasing towards infinity.

To determine the decaying rate, λ_L , the energy conservation method presented in [6] is used. Taking the fluid volume as a unit control volume, there are five streams of heat flux crossing the control volume as follows:

- (I) flux entering the volume at the inlet: $\rho C_P \int_{in} u_x \bar{\theta} dy$;
- (II) flux leaving the control volume at the outlet: $\rho C_P \int_{out} u_x \bar{\theta} dy$;
- (III) streamwise diffusion leaving the control volume at the inlet: $\kappa \int_{in} \left(\frac{\partial \bar{\theta}}{\partial x} \right) dy$;
- (IV) streamwise diffusion entering the control volume at the outlet: $\kappa \int_{out} \left(\frac{\partial \bar{\theta}}{\partial x} \right) dy$;
- (V) heat flux leaving the control volume at the wall surface Ω : $\kappa \int_{\Omega} \left(\frac{\partial \bar{\theta}}{\partial n} \right) ds$.

The energy conservation principles for a steady system, using the above equations, becomes

$$(I) - (II) - (III) + (IV) - (V) = 0 \quad (14)$$

here the subscripts *in* and *out* indicate the inlet and outlet fluxes, n is the direction normal to the wall, C_P is the heat capacity, and κ is the thermal conductivity of the fluid. Setting the inlet at $x = 0$, applying

the periodic relationship in Eqn. 10, and solving for λ_L gives

$$\lambda_L = -\frac{1}{L} \ln \left[1 - \frac{\alpha \int_{\Omega} \left(\frac{\partial \theta}{\partial n} \right) ds}{\int_{in} \left(u_x \bar{\theta} + \alpha \frac{\partial \theta}{\partial x} \right) dy} \right] \quad (15)$$

3. Solver Implementation

The standard **simpleFoam** and **pimpleFoam** solvers are used as the base with the energy equation added in the form of Eqn. 12. Figure 1 shows the steps of the solver, starting with the pressure and momentum equations being solved first utilizing the cyclic constraint on the volumetric average velocity. As with the work of [6,11,12], there is a user specified delay given by the number of iterations, n , before λ_L is calculated, and a delay between subsequent calculations, m . This delay is read into the solver in the **readTransportProperties.H** file and implemented as an if-else statement in the **lambdaEqn.H** file within the source files. The solver keeps track of which iteration is currently being solved for, ΔN , and how many iterations have elapsed since the last calculation of λ_L , ΔM . This delay condition is needed to ensure that all fields have had time to develop.

Initially, a user specified λ_L is used until the number of iterations has reached the desired value, as shown in **lambdaEqn.H**. $\bar{\theta}$ is then calculated from λ_L and Eqn. 12 in **thetaBarEqn.H**. After $\bar{\theta}$ is calculated, the whole field is normalized by the average inlet $\bar{\theta}$ using the **gAverage** function. The temperature fields θ and T can then be calculated readily in **thetaEqn.H**, using Eqn. 11.

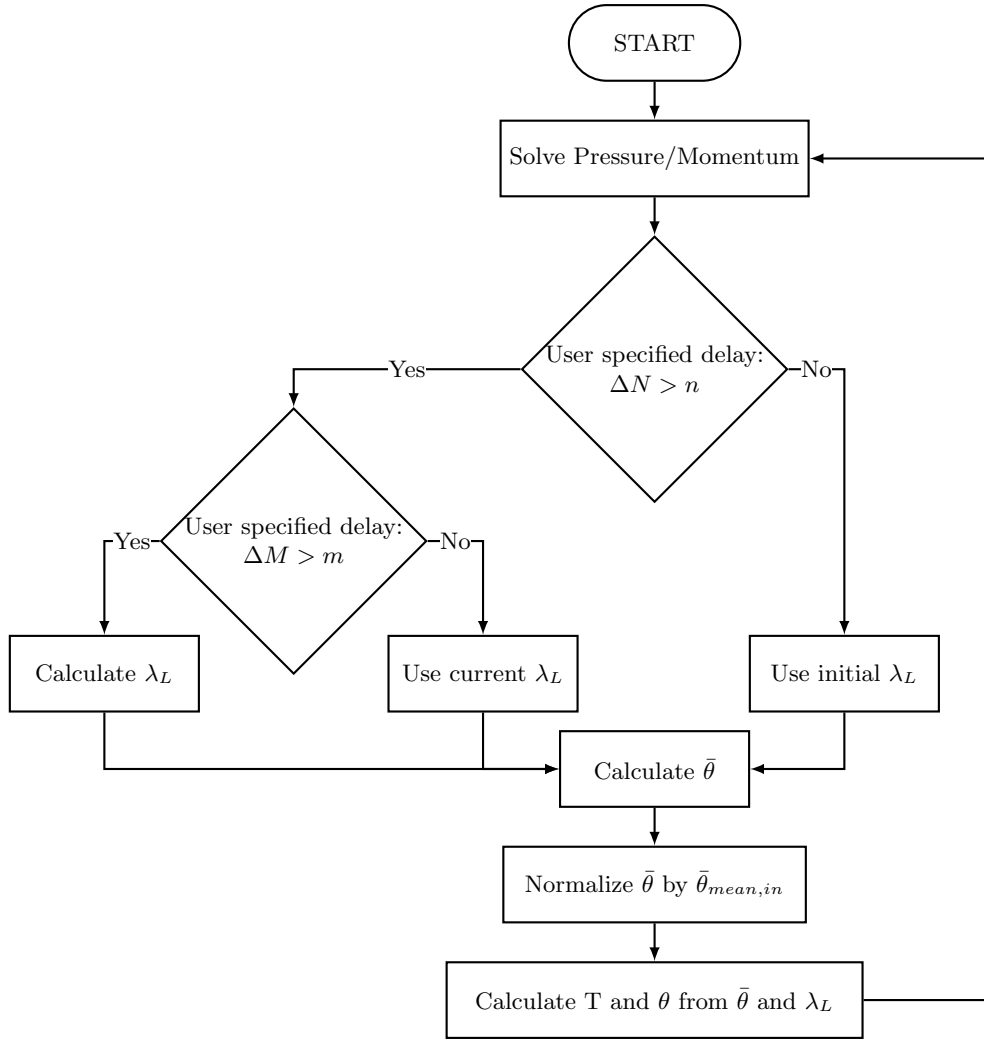


Figure 1. Algorithm used in the cyclic heat transfer solver. Here ΔN signifies the number of iterations the solver has solved the pressure/momentum equation, n is the specified number of iterations used as a delay (2000 for this work), ΔM is the number of iterations in between updating the λ_L value, m is a counter of the number of iterations since the last update of λ_L . The user specified delay is introduced to ensure that the temperature field has developed before calculating λ_L .

4. Validation

To validate the solver, two cases are chosen: 2D Hagen–Poiseuille (HP) flow and 2D wavy channels. The first is chosen as there is an analytical solution and the second as there are many other numerical and experimental studies. First, common simulation parameters for all geometries are presented. Then, results for 2D HP, arc, and sine channels are given and compared to analytical and published results.

4.1. Common Simulation Parameters. Within each validation case, the wall is set to a constant wall temperature $T_w = 293.15$ K and the temperature of the inlet fluid is set to $T_{inlet} = 323.15$ K. All fluid properties are calculated using the pyFluids package [15] for the fluid at atmospheric pressure and film temperature, $T_{film} = 308.15$ K. The average velocity for the cyclic condition, U_{bar} , is set depending on the Reynolds number, which is geometry-dependent and described in the following sections.

Two fluids are considered for this validation, air and water. Table 1 gives the fluid properties used in this work. The parameters of importance when considering fully developed flow and heat transfer in

Table 1. Fluid properties for the two fluids considered in this work.

Property	symbol	Units	Air	Water
Density	ρ	$[kg \cdot m^{-3}]$	1.146	994.033
Dynamic Viscosity	μ	$[kg \cdot m^{-1} \cdot s^{-1}]$	1.893e-05	7.191e-04
Kinematic Viscosity	ν	$[m^2 \cdot s^{-1}]$	1.652e-05	7.234e-07
Prandtl	Pr		0.706	4.834
Heat Capacity	C_P	$[J \cdot kg^{-1} \cdot K^{-1}]$	1006.696	4179.258
Thermal Diffusivity	α	$[m^2 \cdot s^{-1}]$	2.340e-05	1.497e-07
Thermal Conductivity	κ	$[W \cdot m^{-1} \cdot K^{-1}]$	0.027	0.622

channels are the pressure drop, Nusselt number, and friction factor. Pressure drop is directly given by

$$\Delta P = \rho(P_{m,inlet} - P_{m,outlet}), \quad (16)$$

where ρ is the fluid density, $P_{m,inlet}$ is the mean pressure at the inlet patch, and $P_{m,outlet}$ is the mean pressure at the outlet patch. This form is adopted since pressure is reported as kinematic pressure in OpenFOAM. From this, the friction factor is calculated by [16, 17]

$$f = \frac{\Delta P L_c}{2L\rho U_{av}^2}, \quad (17)$$

where L_c is the length scale used in the Reynolds number, L is the length of the domain, ρ is the fluid density, and U_{av} is the mean flow velocity of the module. The Nusselt number is calculated by

$$Nu = \frac{hD_h}{\kappa}, \quad (18)$$

where D_h is the geometry dependent hydraulic diameter, κ is the thermal conductivity of the fluid, and h is the heat transfer coefficient defined as

$$h = \frac{\dot{m}C_p(T_{mc,inlet} - T_{mc,outlet})}{A_{ht}T_{lm}}. \quad (19)$$

Here $\dot{m}$ is the mass flow rate, C_p is the heat capacitance of the fluid, $T_{mc,inlet}$ is the mixing cup temperature at the inlet, $T_{mc,outlet}$ is the mixing cup temperature at the outlet, A_{ht} is the heat transfer surface area, and T_{lm} is the log-mean temperature difference given by

$$T_{lm} = \frac{(T_w - T_{mc,outlet}) - (T_w - T_{mc,inlet})}{\ln[(T_w - T_{mc,outlet}) / (T_w - T_{mc,inlet})]}, \quad (20)$$

where T_w is the wall temperature.

All temperatures are given as mixing cup temperatures, which are normalized by the velocity using

$$T_{mc} = \frac{\sum_{i=0}^N T_i U_{x,i} V_i}{\sum_{i=0}^N U_{x,i} V_i}. \quad (21)$$

Here T is the temperature, U_x is the velocity in the flow direction, V is the volume of the cell, and N is the total number of cells in the domain. The normalization of the velocity profile is defined by

$$U^* = \frac{U}{U_0}, \quad (22)$$

where U_0 is the average inlet velocity, or mean velocity over one module, depending on the geometry. The temperature profile is defined by

$$T^* = \frac{T}{T_0}, \quad (23)$$

where T_0 is the average inlet temperature, or mean temperature over one module. For Hagen-Poiseuille flow, the average velocity and temperature is taken at the inlet, whereas these values are taken over the entire domain.

4.2. Numerical Setup. The OpenFOAM laminar model is used to solve for pressure and momentum. Table 2 provides the numerical schemes used in this work. All schemes were kept the same between the steady and unsteady solver with the only difference being the time scheme. The adjustable time step option is used to keep the Courant number below 1.0 for the unsteady solver.

Table 2. Numerical schemes and their respective term. Schemes are limited to second order, except for time scheme. d_j represents the partial derivative where $1 - 3$ is a Cartesian coordinate direction, i.e., $1 = \partial x$, $2 = \partial y$, $3 = \partial z$.

Term		Scheme	OpenFOAM Entry
time scheme	$\frac{\partial u_i}{\partial t}$	1st order, implicit	Euler
gradients	$\frac{\partial}{\partial d_j}$	2nd order, linear	Gauss linear
advective U	$u_j \frac{\partial u_i}{\partial d_j}$	2nd order, bounded	bounded Gauss limitedLinear 0.2
laplacian	$\frac{\partial^2}{\partial d_j^2}$	2nd order, uncorrected	Gauss linear uncorrected
interpolation	$\phi P \phi L$	2nd order, linear	linear
surface normal gradients	$\frac{\partial}{\partial n}$	No non-orthogonal correction	uncorrected
wallDistance		mesh-wave method	meshWave

4.3. 2D Hagen-Poiseuille Flow. The analytical solution for Hagen-Poiseuille flow is provided in Appendix A. Figure 2 shows the overall simulation domain for one periodic module. For HP flow, the hydraulic diameter, D_h , and characteristic length, L_c , is taken to be $2H$, where H is the height of the domain [18].

The mesh was constructed using the **blockMesh** utility with no grading. A grid convergence study was performed using the method developed by Roache [19,20]. Results are presented in the supplementary document. The following section shows that convergence of the temperature field, specifically λ_L , needs to be considered when using this solver.

4.3.1. λ_L and Velocity Convergence. Care must be taken to ensure that both the velocity and thermal field have reached a fully developed condition, as these happen at different flow times. We make the distinction here between simulation time and run time, with the latter being the amount of time it takes to complete the simulation and the former being the flow time of the simulation. For the purpose of this work, the velocity field is considered fully developed when there is no change between the inlet and outlet velocities. Since the temperature field is calculated directly using the decay constant, λ_L discussed in Section subsection 2.2, the temperature field is considered fully developed when there is no change in λ_L .

Figure 3 and Fig. 4 show the development of λ_L and U_x over simulation time for the steady and unsteady solvers at a range of Reynolds numbers for air, $Pr=0.706$, and water, $Pr=4.486$, respectively.

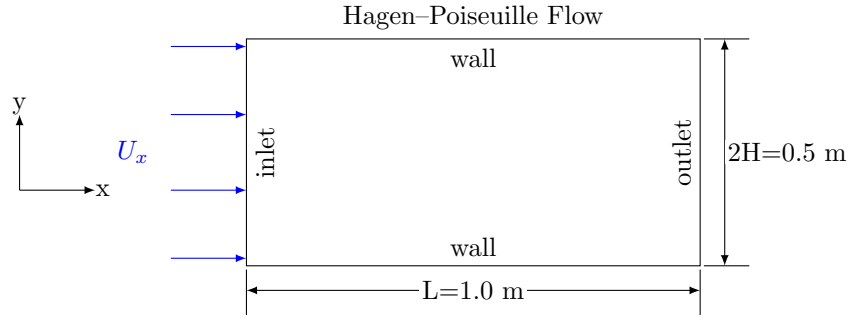


Figure 2. Schematic showing the 2D channel for Hagen-Poiseuille flow and the periodic module taken from the domain.

Figure 3a shows that there is an inverse relationship between λ_L and Reynolds number. A smaller λ_L indicates less influence of the source terms in Eqn. 12 which causes the temperature field to decay less from the inlet to outlet.

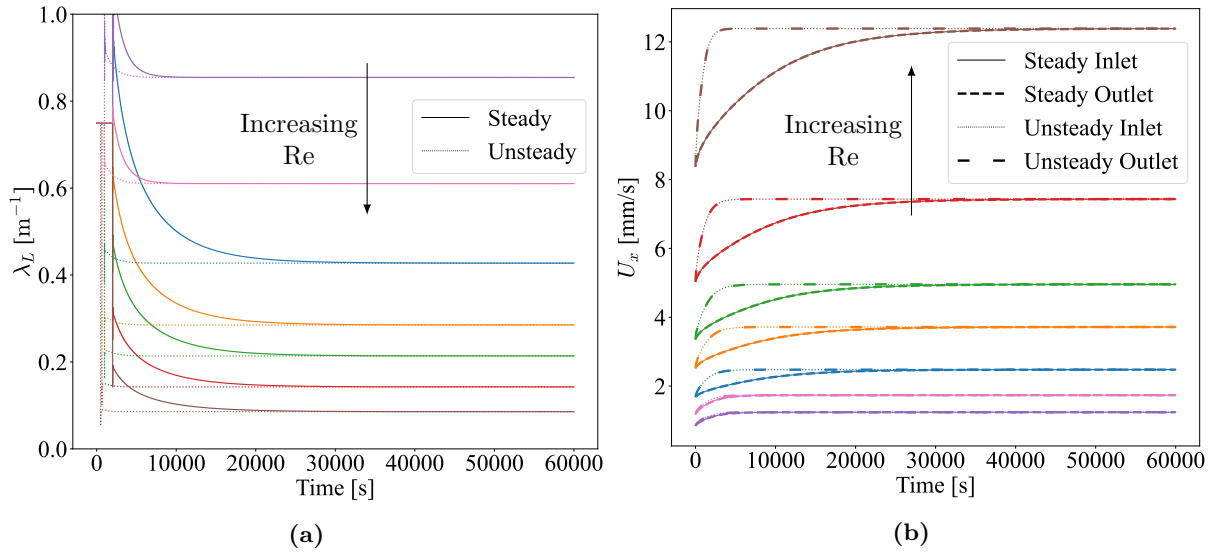


Figure 3. Time for (a) λ_L and (b) inlet / outlet velocities for air at a Prandtl number of 0.706. Both steady and unsteady results are shown. Steady results given per iteration and unsteady results given per flow time in the x-axis. Reynolds numbers are 50, 70, 100, 150, 200, 300, and 500.

Figure 3 and Fig. 4 further show that the λ_L values for water are much lower than that of air at the same Reynolds number. From Fig. 15, this is expected since the thermal diffusivity, α , of water is ≈ 150 times less than that of air. This means that water can move more heat through the system which gives less variation in temperature from the inlet to the outlet, corresponding to a smaller λ_L . Comparing Fig. 3a and Fig. 4a, the unsteady solver for a $Pr=0.706$ reaches the steady-state value much quicker than the steady solver, and the inverse is true for the fluid with $Pr=4.486$. Referring to Fig. 3a and Fig. 3a, water takes 2 to 8 times as long for λ_L to converge compared to air, depending on the Re . These two figures show that care should be taken to examine the convergence of λ_L and velocity for a desired working fluid, as the solver has very different results for different fluid properties.

To add to this idea, the λ_L for the air and water cases are plotted in terms of the Stanton number, $St = \frac{Nu}{RePr}$. According to the analytical temperature field, Eqn. 31, this should be agnostic of the type of fluid and both cases should only be dependent on the Reynolds and Prandtl number. If this condition is true, then the two fluids should fall on the same line, since viscosity is cancelled out in the denominator. Figure Fig. 5 shows that this is the case, which suggests that the decay rate, λ_L , is not dependent on the fluid viscosity, but only on velocity, density, and thermal diffusivity.

4.3.2. Comparison with Analytical Results. Figure 6 gives the temperature field, normalized velocity, and temperature profiles for air at Reynolds numbers of 50 and 500. While only two Reynolds numbers

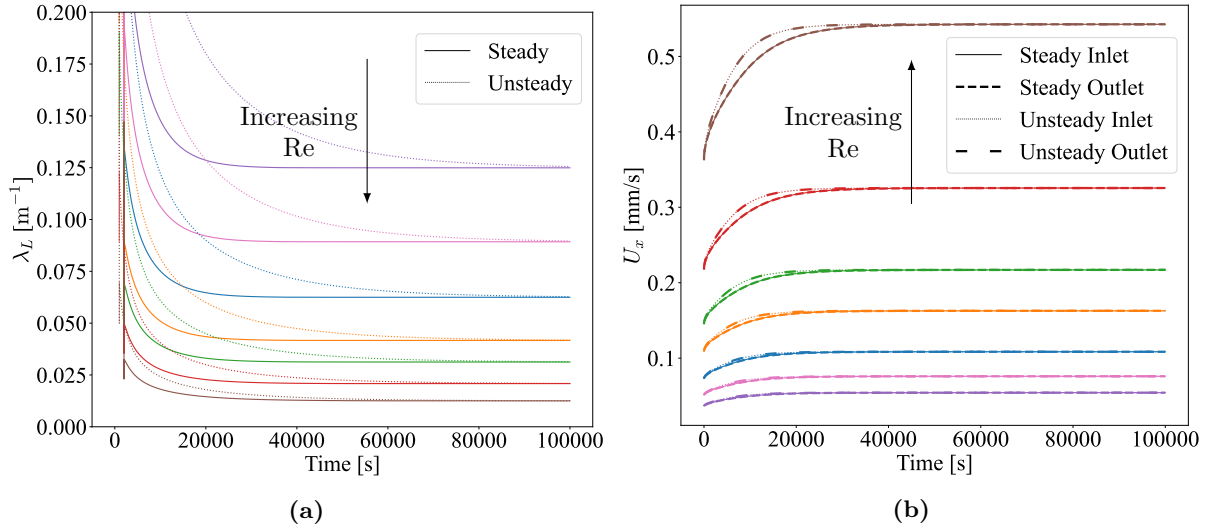


Figure 4. Time for (a) λ_L and (b) inlet / outlet velocities for water at a Prandtl number of 4.834. Both steady and unsteady results are shown. Steady results given per iteration and unsteady results given per flow time in the x-axis. Reynolds numbers are 50, 70, 100, 150, 200, 300, and 500.

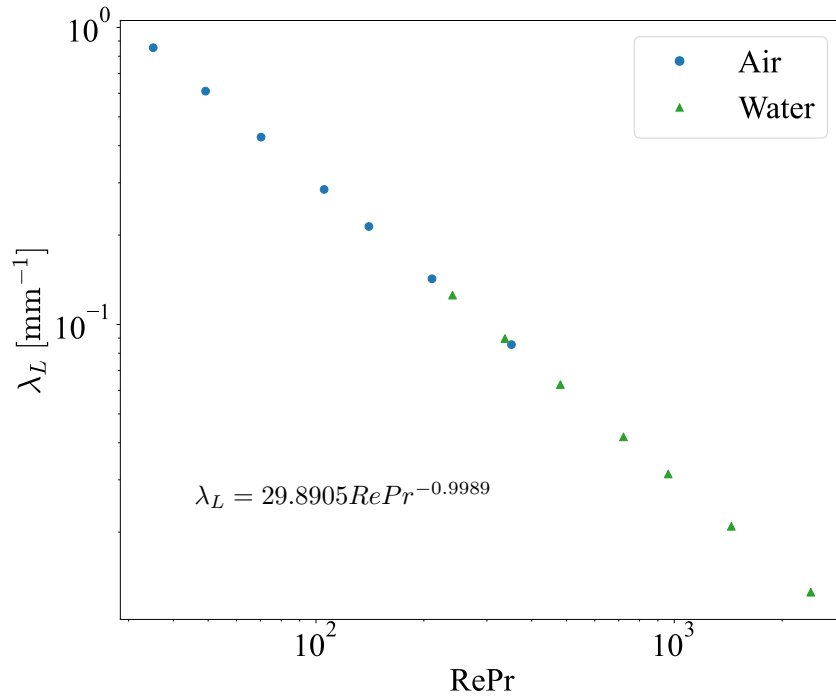


Figure 5. λ_L versus RePr for both the air and water case. Power law fit amplitude and exponent is provided.

are shown here, all Reynolds numbers are provided in the supplementary documentation. The higher Reynolds number case corresponds to a higher simulated velocity with the same viscosity. The inverse relationship of λ_L with Reynolds number, discussed in subsubsection 4.3.1, is shown here when looking at the normalized temperature profiles. There is less variation in the temperature from the inlet to the outlet for the higher Reynolds case. Comparing the temperature isolines from Fig. 6a and Fig. 6b, the temperature gradient from the wall, $\frac{\partial T}{\partial n}$, for the $\text{Re}=50$ case is higher than for $\text{Re}=500$. Furthermore, the temperature gradient in the x direction $\frac{\partial T}{\partial x}$ is also higher for the $\text{Re}=50$ case. Referring to Eqn. 15, this suggests that the velocity in the x-direction is the dominating term for the decay rate since λ_L decreases with increasing Reynolds number.

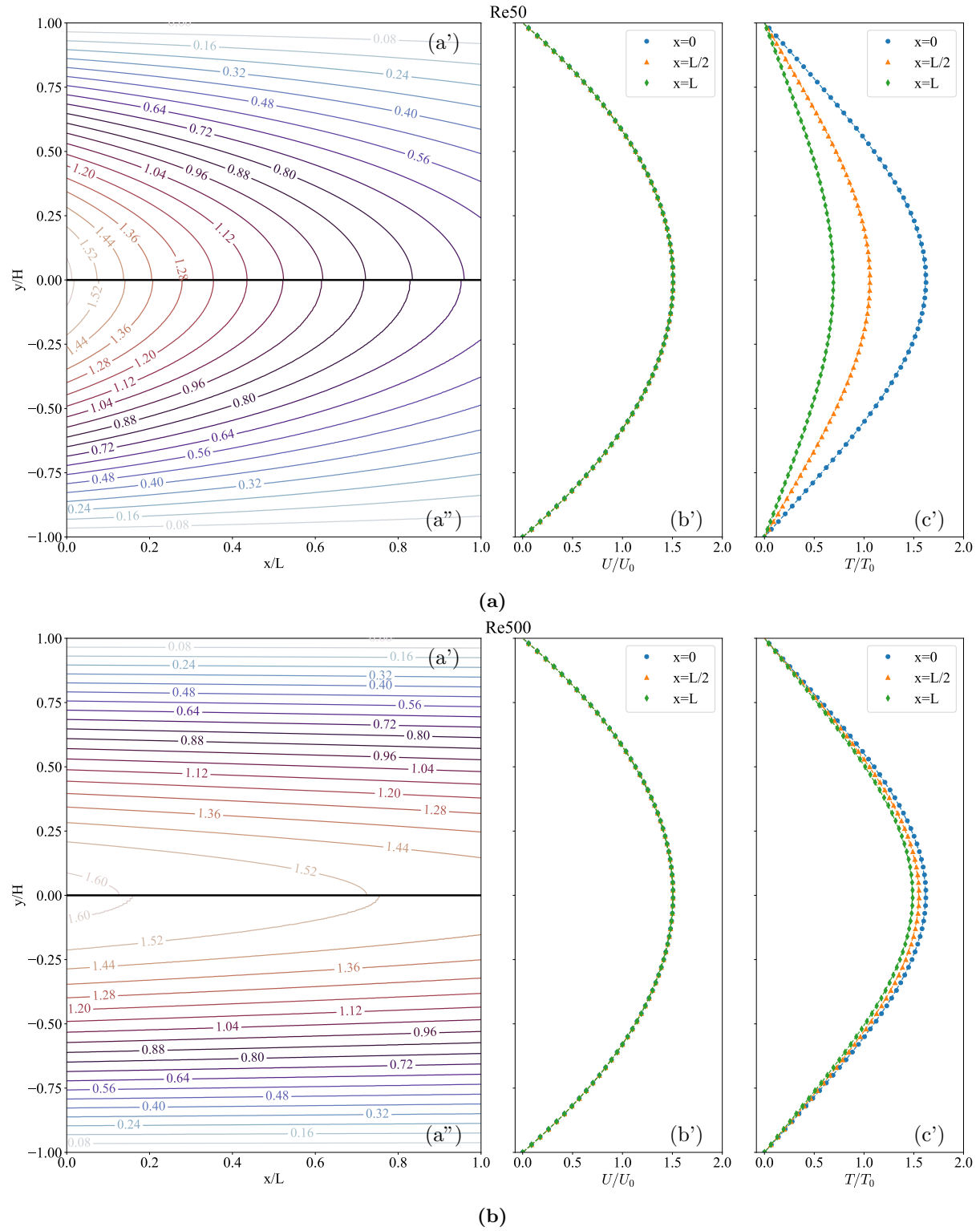


Figure 6. (a) Reynolds 50 and (b) Reynolds 500 steady solver cases for air. (a') analytical and (a'') simulated temperature field. The normalized (b') velocity and (c') temperature profile are given at the inlet ($x = 0$), halfway ($x = L/2$), and outlet ($x = L$).

To further validate the solver for the two fluids, the Nusselt number, Nu , is calculated using Eqn. 18, and the results are shown in Fig. 7. Both solvers show good agreement with the Nu over the range of Reynolds numbers for both fluids. There is much more scatter for the water case, but this scatter is all within 5% of the analytical value. The normalized analytical and simulated profiles, as shown in Fig. 6, for all the tested Reynolds number are compared using the root-mean-square-error (RMSE). This error

is calculated using

$$RMSE = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{N}}, \quad (24)$$

where y_i is the simulated value for observation i , $\hat{y}_i$ is the analytical value for observation i , and N is the number of observations, 100 for this work. Each profile is linearly interpolated to ensure that the analytical and simulated points on each profile match each other. Table 3 and Tab. 4 give the RMSE for air and water, respectively. Both steady and unsteady solver cases are represented and the values are multiplied by 100 for clarity. In most cases, the steady solver provides a closer profile to the analytical solution, but the differences between the solvers are minor. Important here is that each profile is normalized to ≈ 1 so these values indicate that the error is less than 1%.

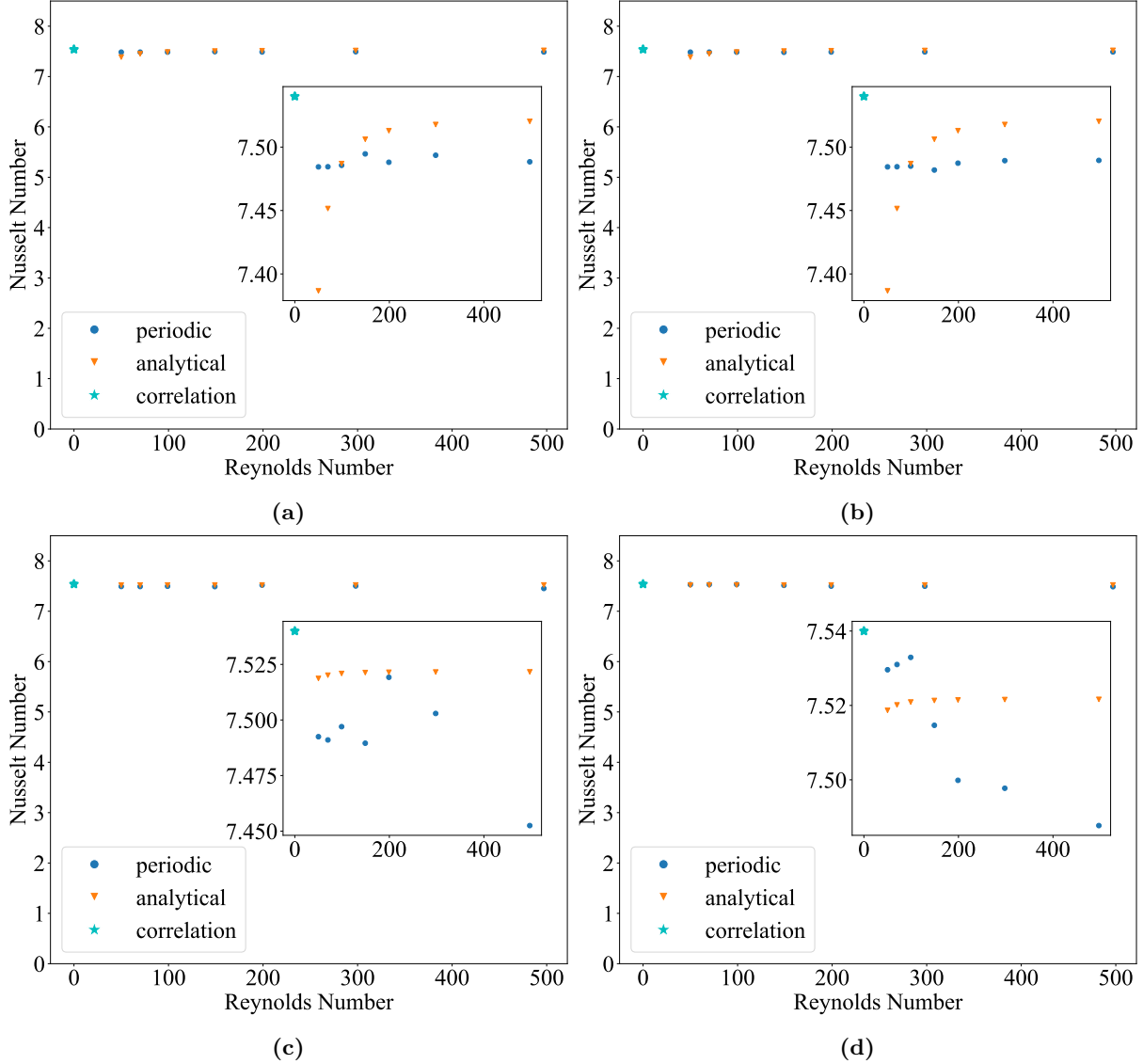


Figure 7. Steady and unsteady Nusselt number versus Reynolds number for cases (a)(b) air with Prandtl number 0.706 and (c)(d) water with Prandtl number 4.834. Nusselt correlation is a constant value of 7.54 in the laminar regime for Hagen-Poiseuille flow with constant wall temperature boundary condition [17].

It was found that the decay rate, Eqn. 15, is sensitive to low Reynolds numbers. At any Reynolds number below ≈ 50 , the denominator of Eqn. 15 is smaller than the numerator causing a solver error due to taking the natural log of a negative number. This indicates that the heat flux at the inlet is smaller than the heat flux from the walls. As the Reynolds number decreases, the system approaches a state of conduction instead of forced convection, where all the heat flux is from the walls with little contribution

Table 3. Root-mean-square-error of temperature profiles for steady and unsteady solver. RMSE reported here is multiplied by 100 for readability. Air is the working fluid with a Prandtl number of 0.706. The first column (highlighted in gray) under each Reynolds number is steady solver RMSE and the second column is the unsteady solver RMSE. T is the temperature profile at the inlet ($x = 0$), midway ($x = L/2$) and outlet ($x = L$). Velocity profiles have an RMSE of nearly zero, so are not shown on this table.

Profile	Reynolds Numbers													
	50.0		70.0		100.0		150.0		200.0		300.0		500.0	
T ($x = 0$)	0.090	0.090	0.047	0.047	0.041	0.041	0.057	0.047	0.065	0.062	0.077	0.069	0.093	0.093
T ($x = L/2$)	0.068	0.068	0.201	0.201	0.269	0.273	0.236	0.196	0.211	0.217	0.179	0.182	0.139	0.141
T ($x = L$)	0.462	0.462	0.084	0.084	0.103	0.105	0.131	0.120	0.155	0.156	0.135	0.137	0.129	0.127

Table 4. Table of root-mean-square-error of temperature profiles for steady and unsteady solver. RMSE reported here is multiplied by 100 for readability. Water is the working fluid with a Prandtl number of 4.486. The first column (highlighted in gray) under each Reynolds number is steady solver RMSE and the second column is the unsteady solver RMSE. T is the temperature profile at the inlet ($x = 0$), midway ($x = L/2$) and outlet ($x = L$). Velocity profiles have an RMSE of nearly zero, so are not shown on this table.

Profile	Reynolds Numbers													
	50.0		70.0		100.0		150.0		200.0		300.0		500.0	
T ($x = 0$)	0.079	0.300	0.082	0.306	0.095	0.306	0.097	0.308	0.311	0.308	0.311	0.311	0.309	0.312
T ($x = L/2$)	0.160	0.330	0.143	0.333	0.129	0.334	0.125	0.324	0.112	0.321	0.114	0.317	0.170	0.319
T ($x = L$)	0.129	0.279	0.126	0.286	0.113	0.288	0.109	0.303	0.104	0.309	0.112	0.314	0.174	0.311

from the fluid flow. Interestingly, the unsteady solver was able to achieve a lower Reynolds number than the steady solver, but was not able to resolve λ_L below a value of $\text{Re} \approx 45$.

A further observation can be seen in Fig. 8 when comparing air and water at the same Reynolds number. The data shows that there is more heat transferred from the inlet to the outlet for air than for water. Water has a thermal diffusivity ≈ 150 times less than air, meaning water will conduct heat less quickly than air. This explains the more shallow parabolic shape of the temperature fields in Fig. 8a when compared to Fig. 8b. Furthermore, this is an example of why the heat decay rate, λ_L , of water is much less than that of air.

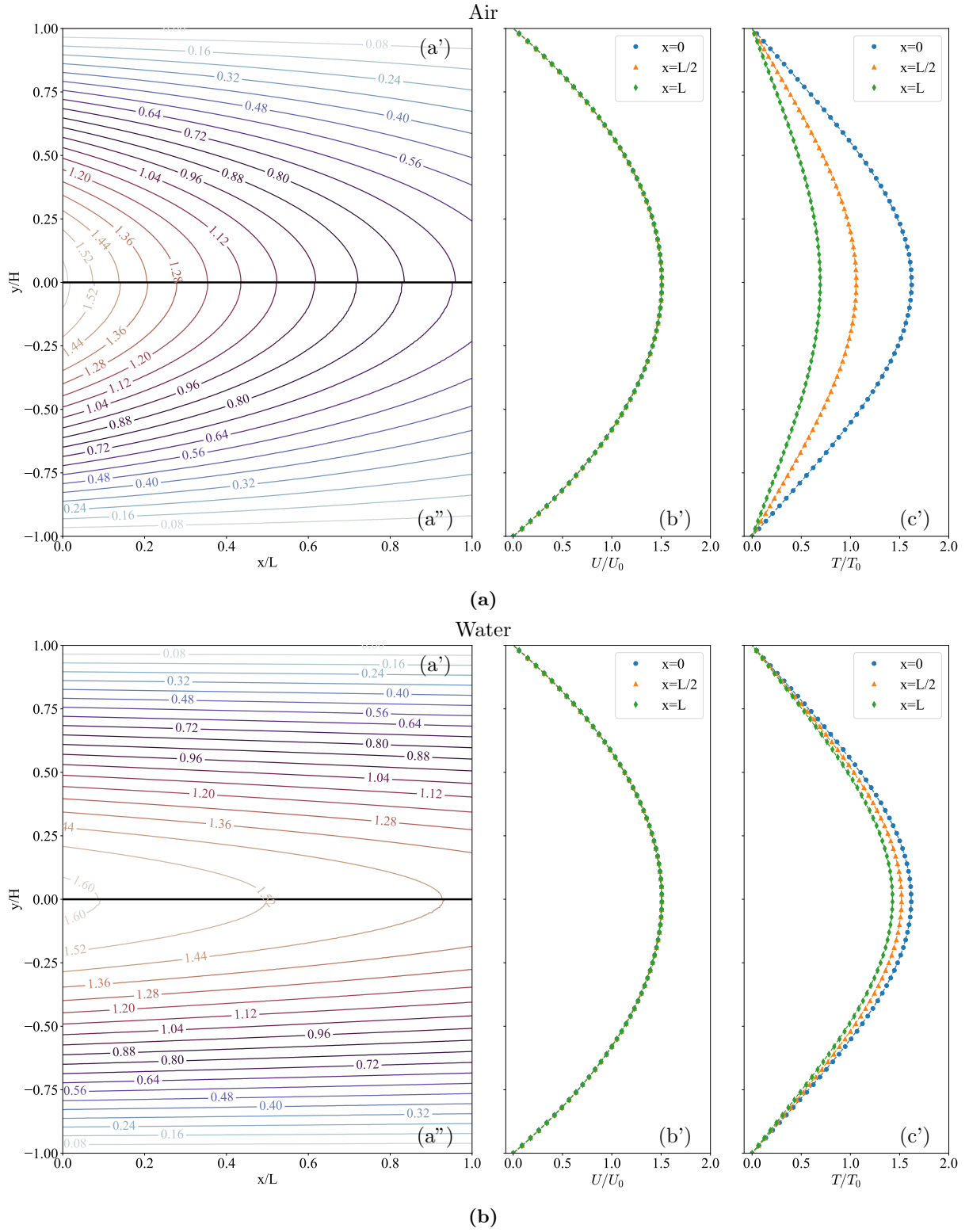


Figure 8. (a) Reynolds 50 for air and (b) Reynolds 50 for water using the unsteady solver. (a') analytical and (a'') simulated temperature field. The normalized (b') velocity and (c') temperature profile are given at the inlet ($x = 0$), halfway ($x = L/2$), and outlet ($x = L$).

4.4. 2D Wavy Channel. A more complex geometry that has been investigated using periodic solvers is the wavy channel [3, 12, 13, 21, 22]. Two types of channels are presented in the literature: sinusoidal and arc, schematically shown in Fig. 9. The wavy channels in this work are defined by the following two independent variables: H_{min}/H_{max} and L/a , where H_{min} is the minimum channel height, H_{max} is the maximum channel height, L is the length of the channel, and a is the amplitude of the height function. The height can then be defined by the function

$$y_{sinusoidal} = (H_{min}/2) + 2a \sin^2(\pi x/L), \quad (25)$$

$$y_{arc} = (H_{min}/2) + 2a \sin(\pi x/L), \quad (26)$$

where the subscript on y signifies the type of channel. For this work, H_{min}/H_{max} is 0.3 and $L/a = 8$. The characteristic length scale, L_c , for the Reynolds number and friction factor is taken as the average height of the channel

$$L_c = H_{avg} = \frac{H_{max} + H_{min}}{2}, \quad (27)$$

where H_{max} and H_{min} are the maximum and minimum height, respectively.

Figure 9 shows the domain of the two channel geometries and the corresponding mesh. The length scale for the Nusselt number is double the average channel height in order to be consistent with the definition provided in literature [11, 21–25]. This length scale does not correspond to the hydraulic diameter, as it would be a solution to an elliptic integral with the key parameters: H_{max} , H_{min} , and L . The mesh was constructed using the **blockMesh** utility and a python script to define the edges.

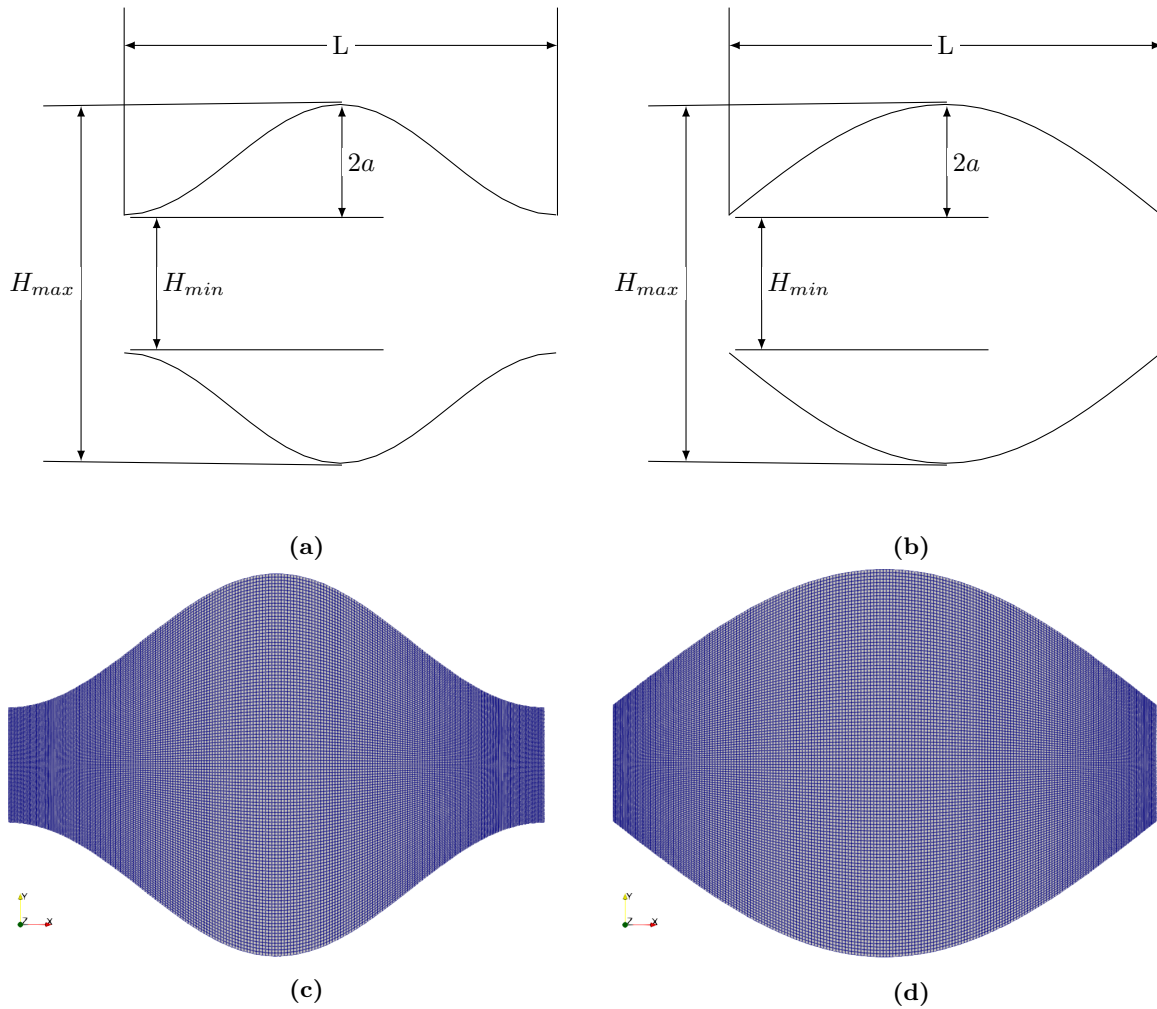


Figure 9. Schematic and corresponding mesh for 2D (a)(c) sinusoidal and (b)(d) arc case for fluid flow through a wavy channel used in this validation.

4.4.1. λ_L and Velocity Convergence. As with the Hagen-Poiseuille flow case, the convergence of the temperature and velocity fields are analyzed. Figure 10 and Fig. 11 show λ_L and two U_x sampling points at the center line of the inlet and outlet. For the arc case, Fig. 10b shows that the flow transitions from the steady regime to the unsteady regime after $Re \approx 70$. This is consistent with the findings of Ničeno and Nobile [24] where the unsteady regime for the arc shaped channel was found to be at $Re = 84$. The sine shaped channel, Fig. 11b, shows that the transition to the unsteady regime occurs after $Re = 150$, which is consistent with the value of $Re = 175$ found by Ničeno and Nobile. This is further validated by Wang and Vanka [23], which predicted the unsteady state to be $Re = 167$, and Nishimura *et al.* [25], which qualitatively found this transition when increasing from $Re = 100$ to $Re = 300$.

An interesting phenomena occurs to the steady state solver in this unsteady region. Referring to Fig. 10 and Fig. 11, it is seen that the steady solver reaches a steady-state value, but then diverges at a sufficiently large flow time. This suggests that due to the flow being unsteady at this Reynolds number, the formulation is too stiff for the steady-state solver. For this reason, all data reported for these cases are from the unsteady solver. For results in the unsteady flow regime, after a sufficiently large amount of simulation time has passed, the last 1000 data points from the unsteady solver are averaged for the calculation of Nu and f .

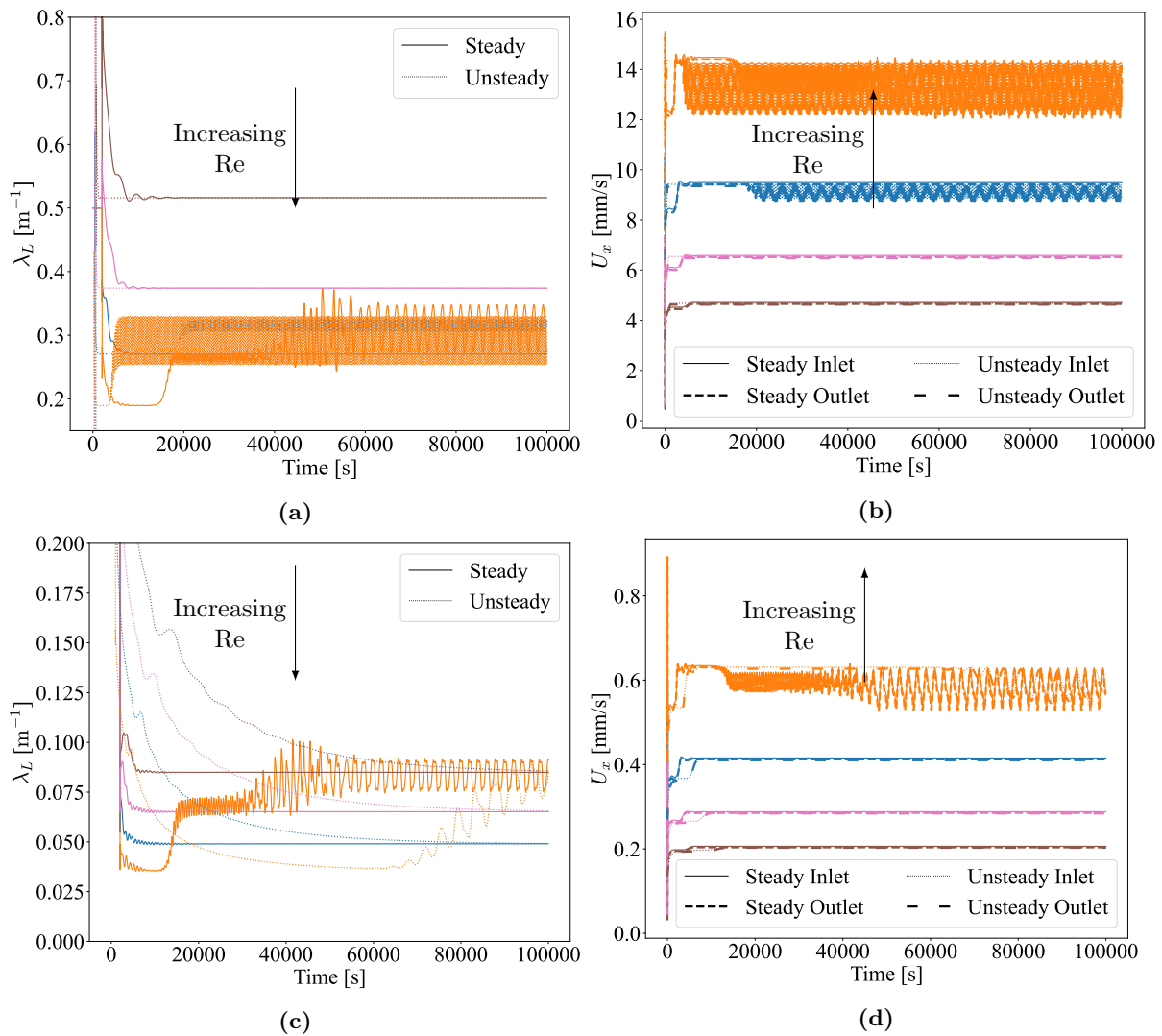


Figure 10. Arc geometry (a) λ_L , (b) U_x convergence for air and (c) λ_L , (b) U_x convergence for water. Reynolds numbers are 50, 70, 100, and 150

To further illustrate the effect of each solver in the unsteady flow regime on the temperature field, temperature and velocity profiles, Fig. 12 shows the steady and unsteady case for the sine geometry at $Re \approx 200$. As expected, the unsteady profiles are more uniform and symmetrical. Furthermore, there is a drastic influence of these disturbances on the overall temperature field. These results show that caution must be taken in terms of which solver is used, which flow regime exists, how many time steps are used,

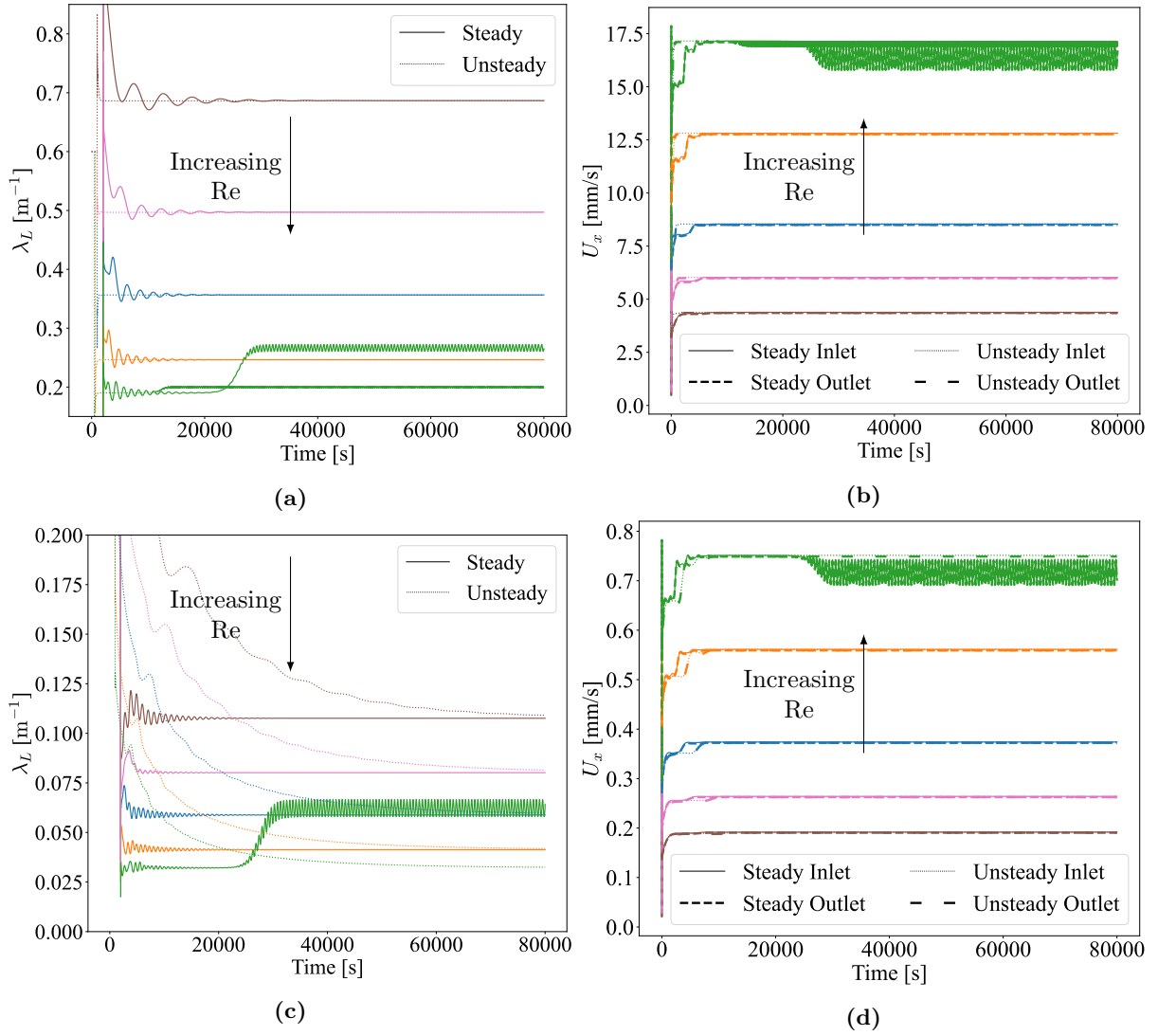


Figure 11. Sine geometry (a) λ_L , (b) U_x convergence for air and (c) λ_L , (d) U_x convergence for water. Reynolds numbers are 50, 70, 100, 150, and 200.

and how converged the solution is when simulating higher Reynolds numbers with these more complex geometries.

As with the plane channel flow in subsubsection 4.3.1, the simulation time is quantified for both steady and unsteady cases in the supplementary document. It is shown that in the transition from steady to unsteady flow regimes, the simulation time steps more than double. This is seen in the arc-channel case from $Re \approx 70$ to $Re \approx 100$ and in the sine-channel case from $Re \approx 150$ to $Re \approx 200$. As previously discussed, this is the range where the field variables begin to oscillate around some average value, which indicates that more solver iterations were needed to solve for the field variables.

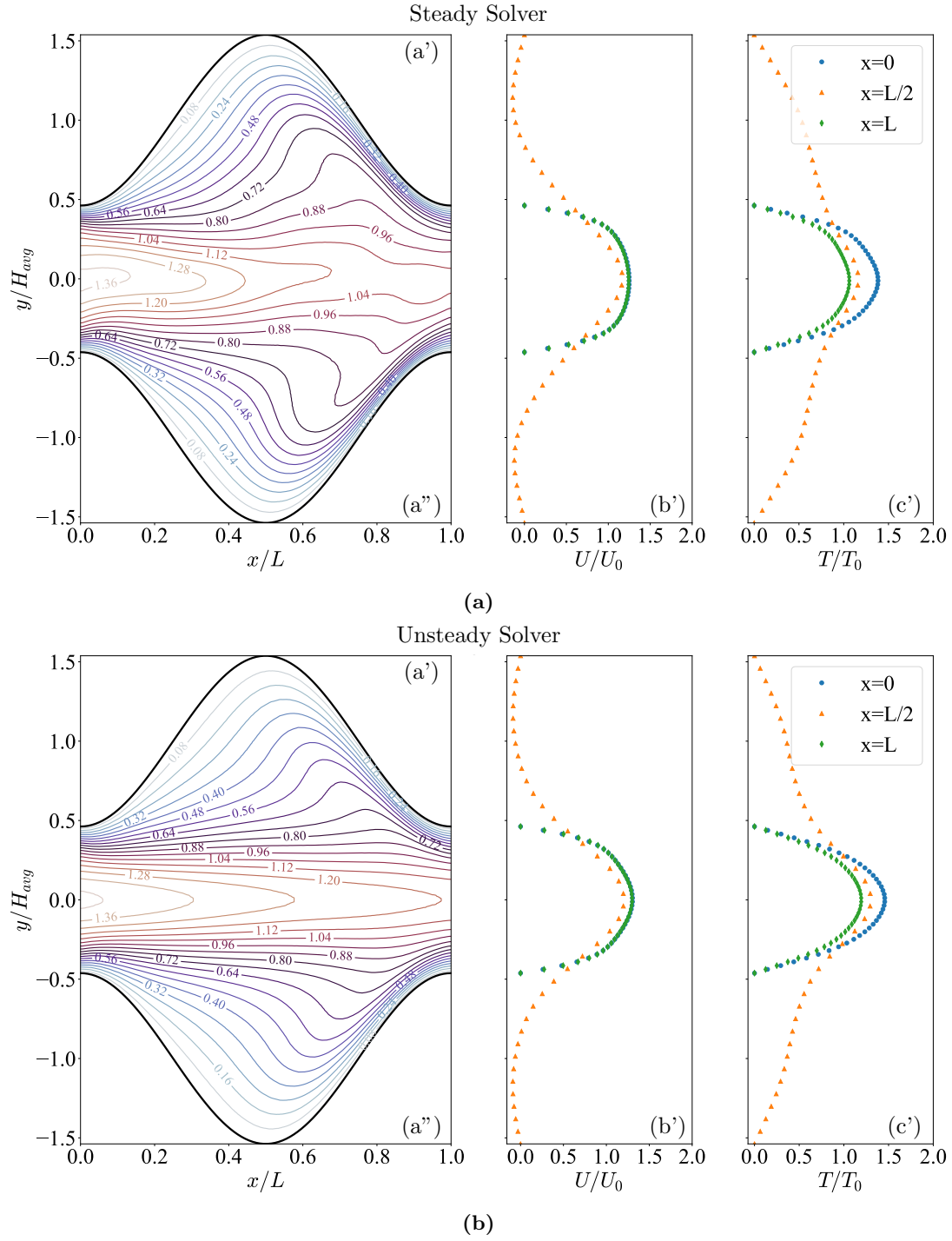


Figure 12. Reynolds 200 using the (a) steady solver and (b) unsteady solver cases for the sine geometry using air as the fluid. (a') analytical and (a'') simulated temperature field. The normalized (b') velocity and (c') temperature profile are given at the inlet ($x = 0$), halfway ($x = L/2$), and outlet ($x = L$). Both simulations (a) and (b) are at $Re \approx 200$.

4.4.2. Comparison with Published Works. The published literature on wavy or corrugated ducts focuses on air, $Pr=0.7$, for a range of Reynolds numbers [11, 21–25]. The dominant published geometry is the sine channel, with a few groups performing numerical simulations on both classes of geometry [21, 24]. Fig. 13 gives a comparison of the Nusselt and friction factors compared to published data.

For the arc channel case, in the steady flow regime, Fig. 13a and Fig. 13c show there is agreement in the shape and trend between the simulation and the data from Ničeno and Nobile [24]. After the flow becomes unsteady, the results start to diverge from the published data, but still follow the same shape and trend of transitioning from a constant value to an exponential value. Interestingly, the simulated data is between the values provided by Ničeno and Nobile and Bahaidarah *et al.* [21]. Bahaidarah performed the numerical analysis on a channel of six modules, and the value of the 4th module was used as a data point. This possibly contributes to the discrepancy between the data sets.

The friction factor for the arc-channel, Fig. 13c shows consistently lower results than the other data sets. The data suggests that this pressure gradient is consistently lower for the same U_{av} as published data. Interestingly, friction factor reaches a constant value at higher Reynolds numbers, also seen in the data from Ničeno and Nobile, which suggests the present work follows the same trend.

For the sine channel case, Fig. 13b and Fig. 13d show agreement with the Nusselt number in the steady region. After flow becomes unsteady, the present work shows a similar trend with published data, and is in the middle of the spread of available data. A possible explanation for this is in how the data was handled in the unsteady region. For the present work, the last 1000 time steps were used to average the velocity and temperature fields, whereas the procedure of the published works in the unsteady regime is unclear. The choice of post processing procedure in this region, with oscillating field variables, as shown in Fig. 10 and Fig. 11, would have large effect on the Nusselt number. Therefore, we are not confident of the results from the simulations in this region.

Figure 13d shows the present work is closest with the experimental results of Nishimura *et al.* [25] and the numerical results of Wang and Vanka [23], but is consistently lower than all historical literature, as with the arc-channel case. For the sine channel case, velocity profiles were reported which are used as a further validation. Fig. 14 shows the data from the sine channel at $Re \approx 100$ and $Re \approx 400$ with the velocity profiles from the numerical results of Bahaidarah *et al.* [21] and Harikrishnan and Tiwari [11]. The velocity profiles at the mid line ($x = L/2$) show good agreement with some discrepancies at the minima and maxima. This suggests that the current unsteady solver resolves the correct velocity profile.

4.4.3. Summary of Simulated Air and Water Results. A summary of only the air and water results for both configurations of wavy channels is provided here. Figure 15 gives the normalized velocity and temperature profiles for both channels and fluid. Figure 15a indicates that there is more flow separation for the water case than for air at the same Reynolds number in each channels respective unsteady regime. This then translates to a markedly different normalized temperature profile, as seen in Fig. 15b. These results suggest that the data presented in the unsteady regime, even though it shows agreement with published works, should be carefully scrutinized.

Figure 16 gives the Nusselt number and friction factor for each channel and fluid combination. In the steady regime, both arc and sine channels with air as the fluid are consistent with Ničeno and Nobile [24]. The arc case, in particular, has a change in slope when transitioning to the unsteady regime, which is consistent with Ničeno and Nobile and the experimental results of Nishimura *et al.* [25]. Interestingly, water has an increased Nu in the sine channel case and a decreased Nu for the arc channel, in the steady flow regime. After a certain Reynolds number, the water case for both channels show a similar behavior of increasing drastically. Comparing this behavior to the friction factor in Fig. 16b, the jump in Nusselt number corresponds to a jump in the friction factor, i.e. the pressure gradient. This is further supported by the drastic change in normalized velocity profiles, shown in Fig. 15a.

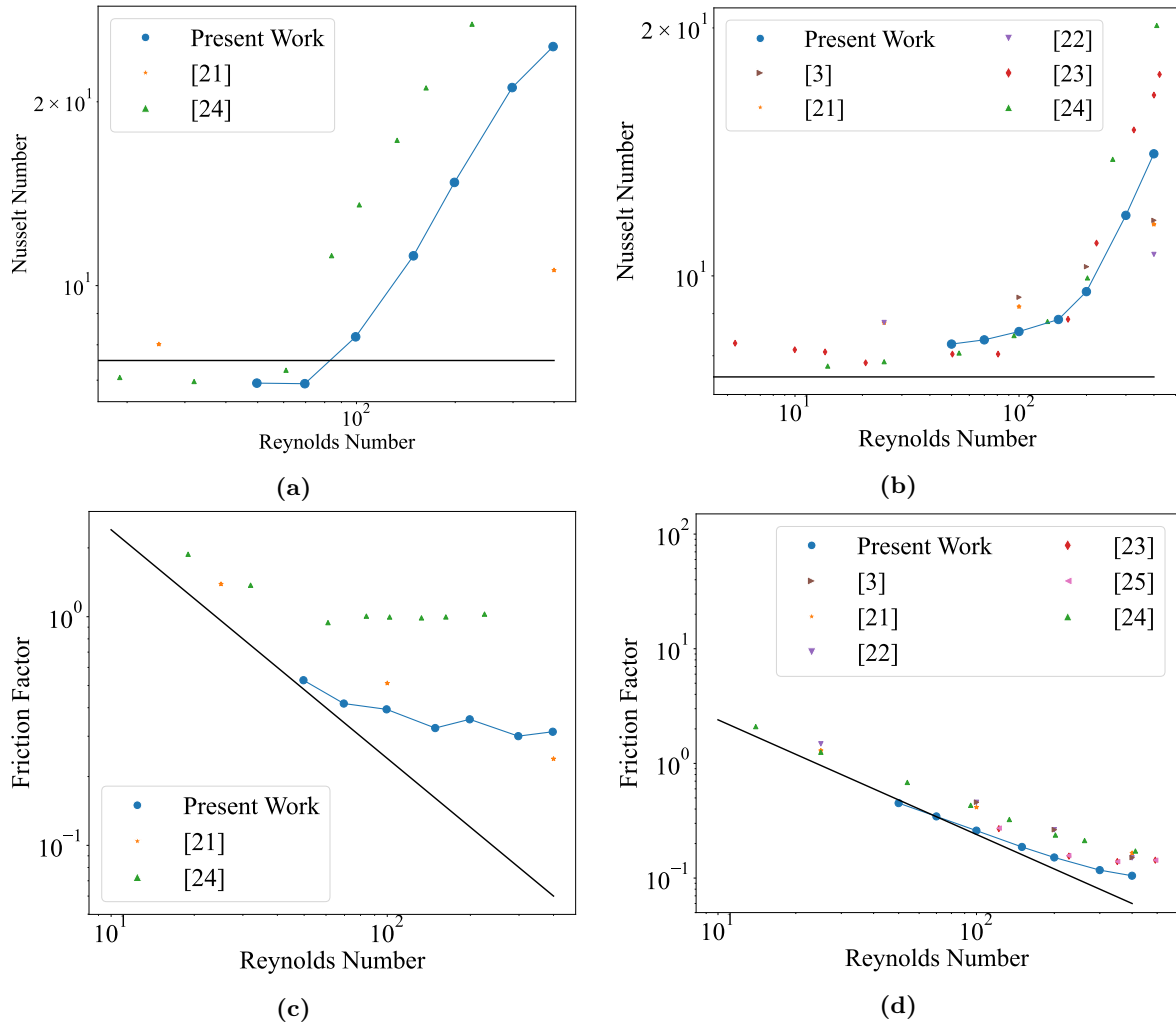


Figure 13. Nusselt number versus Reynolds number for (a) arc and (b) sine channel geometries. Friction Factor versus Reynolds number for (c) arc, and (d) sine channel geometries. Solid black line represents the experimental correlation for plane channel flow.

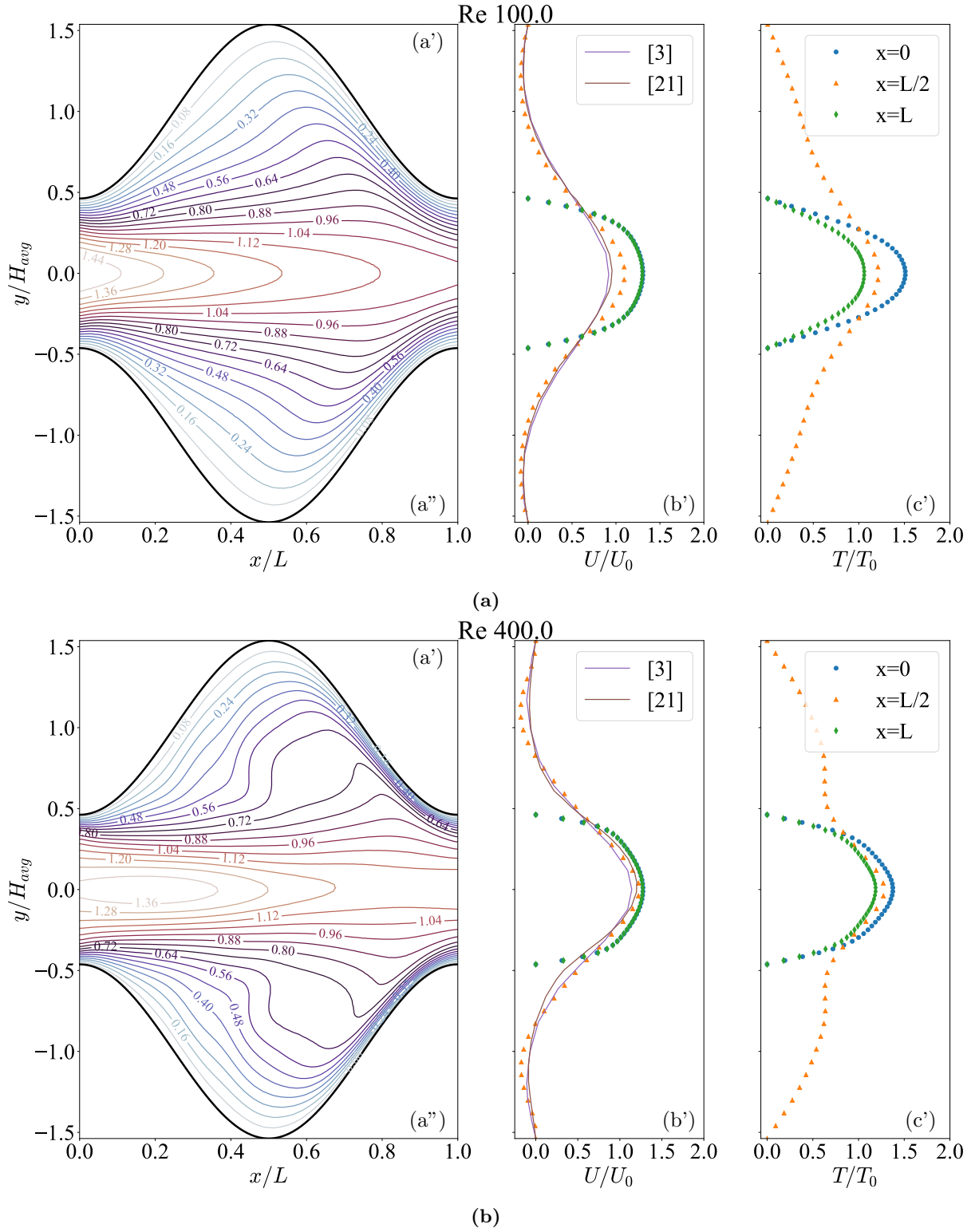


Figure 14. Unsteady results for (a) $Re=100$ and (b) $Re=400$ cases for the sine channel using air as the fluid. (a') analytical and (a'') simulated temperature field. The normalized (b') velocity and (c') temperature profile are given at the inlet ($x=0$), halfway ($x=L/2$), and outlet ($x=L$).

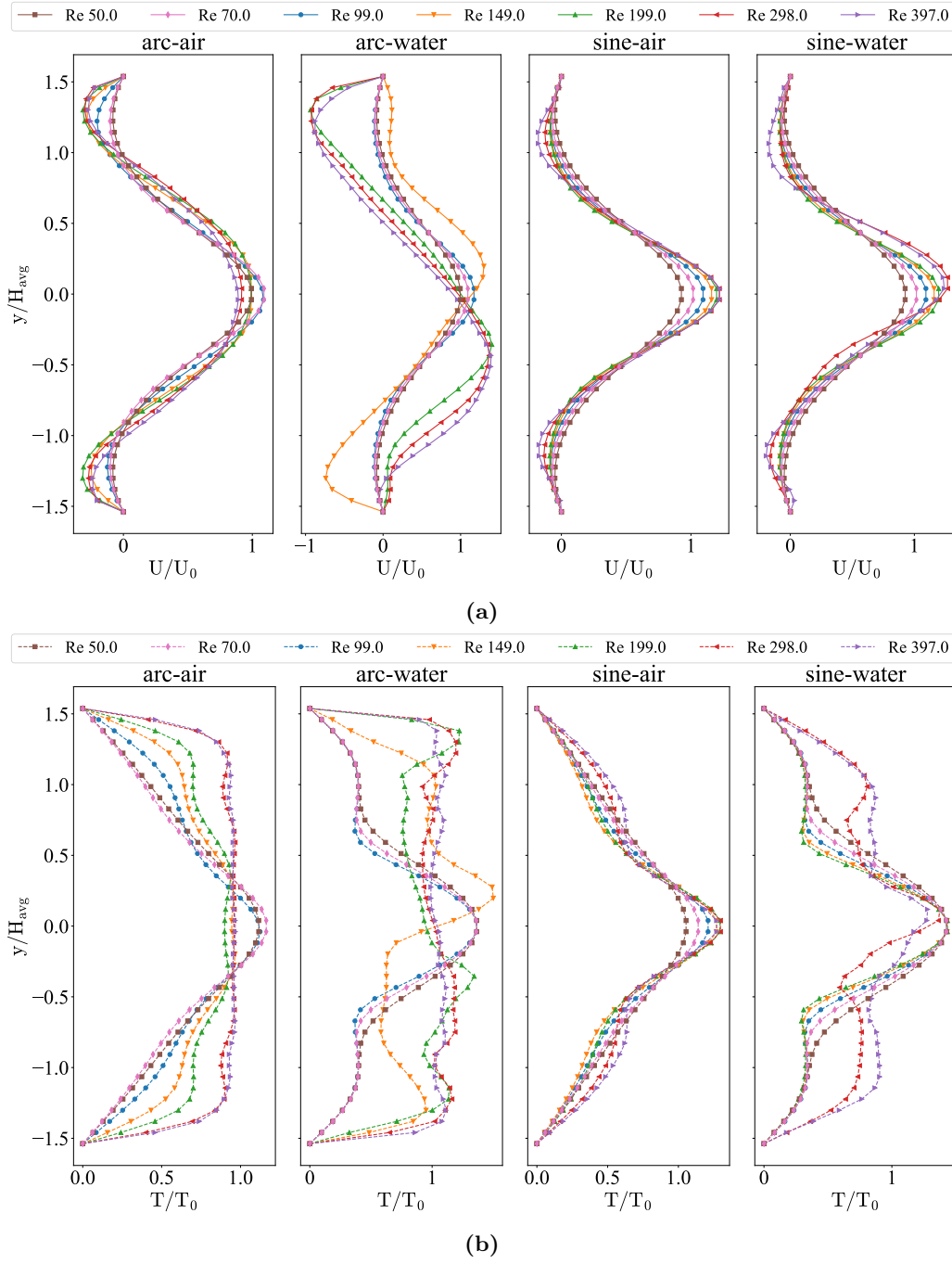


Figure 15. Normalized (a) velocity and (b) temperature profiles for each channel and Reynolds number midway through the channel ($x = L/2$). Channel and fluid are provided as titles for every plot. All data presented are results from the unsteady solver.

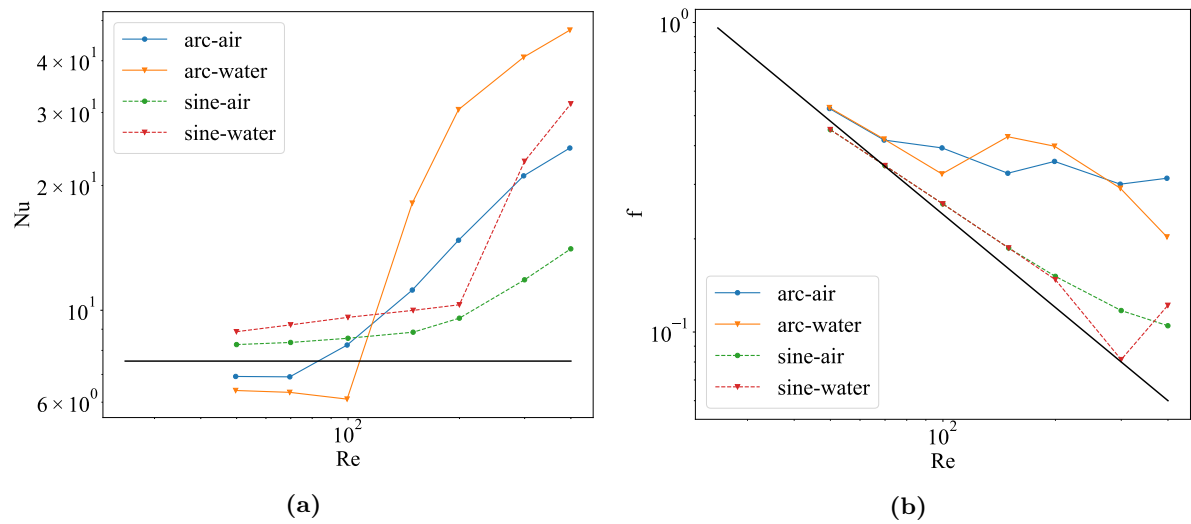


Figure 16. (a) Nu and (b) friction factor versus Reynolds number for each channel and fluid combination. All data presented is results are from the unsteady solver. Solid black line represents the experimental correlation for plane channel flow.

5. Conclusion

In this work a heat transfer solver for cyclic boundary conditions in the laminar regime is presented. This solver includes both a steady solver based on the SIMPLE algorithm and an unsteady solver based on the PIMPLE algorithm. Two validation cases, Hagen-Poiseuille flow and Wavy channels, are explored to characterize the behavior of both types of solvers. Hagen-Poiseuille flow is an important comparison as there is an analytical solution so the velocity and temperature fields can be directly compared. The two wavy channels are provided to characterize the solver's behavior with a more complex geometry. When simulating Hagen-Poiseuille flow, the results fall within 1% of the analytical solution, and discrepancies within 5% when referencing the experimental Nusselt correlation. When plotting λ_L compared to Reynolds and Prandtl numbers, it is found that both fluids simulated fall on the same line, confirming these are appropriate dimensionless variables for this system.

In addition to the Hagen-Poiseuille case, the study delved into wavy channels, aiming to evaluate the solver's performance in more complex geometries. The simulations exhibited acceptable agreement when compared to published data, particularly for the more laminar flow regimes. For the sine wave geometry, the simulated data followed experimental and simulated cases, deviating by approximately 20% from published data, while the arc geometry showed a variation of up to 60%, partly attributed to increased oscillations in flow and temperature profiles. Nevertheless, these findings confirm that the cyclic heat transfer solver aligns with prior research. However, it is crucial to ensure convergence of parameters such as the decay rate (λ_L), especially when employing different working fluids, as water and air demonstrate very different convergence times for the temperature field.

Future developments for this solver include incorporating an option for constant wall flux in addition to constant wall temperature. Work will be done to integrate the constant wall flux model into this solver with an option to use one or the other. Furthermore, there may be cases where performing conjugate heat transfer is more appropriate than fixing the wall temperature or flux. In these cases, the solver will be extended to the framework of OpenFOAM's conjugate heat transfer solvers. Lastly, a multi-scaling approach will be developed for select structures to scale from a periodic unit cell to the larger domain.

Acknowledgements

This work was made possible by the use of the RCC facilities at the University of Canterbury. The author(s) wish to acknowledge the use of New Zealand eScience Infrastructure (NeSI) high performance computing facilities, consulting support and/or training services as part of this research. New Zealand's national facilities are provided by NeSI and funded jointly by NeSI's collaborator institutions and through the Ministry of Business, Innovation & Employment's Research Infrastructure programme. URL <https://www.nesi.org.nz>.

Author Contributions: Conceptualisation, M.C. and D.H.; methodology, M.C.; software, M.C.; validation, M.C. and D.H.; formal analysis, M.C. and D.H.; investigation, M.C.; resources, D.H.; data curation, M.C.; writing—original draft preparation, M.C.; writing—review and editing, M.C. and D.H.; visualisation, M.C.; supervision, D.H.; project administration, D.H.; funding acquisition, D.H. All authors have read and agreed to the published version of the manuscript.

Appendix A. Analytical solution for fully developed flow between parallel plates with constant wall temperature

Consider two parallel plates with a separation H , shown in Fig. 17.

For a fully developed and steady flow, the streamwise velocity field is [14,17]

$$u_x(y) = \frac{3}{2}U_0 \left[1 - \left(\frac{2y}{H} \right)^2 \right], \quad (28)$$

where the flow velocity is u_x and the transverse velocity is $u_y = 0$. U_0 is the mean velocity, y is the transverse coordinate that measure from the centerline $y = 0$. For this calculation, only the domain of $y = [0, \frac{H}{2}]$ is considered as the result is symmetric about $y = 0$. Substituting Eqn. 28 into the energy equation Eqn. 8, and assuming steady state, yields

$$\frac{3}{2}U_0 \left[1 - \left(\frac{2y}{H} \right)^2 \right] \frac{\partial T}{\partial x} = \alpha \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right). \quad (29)$$

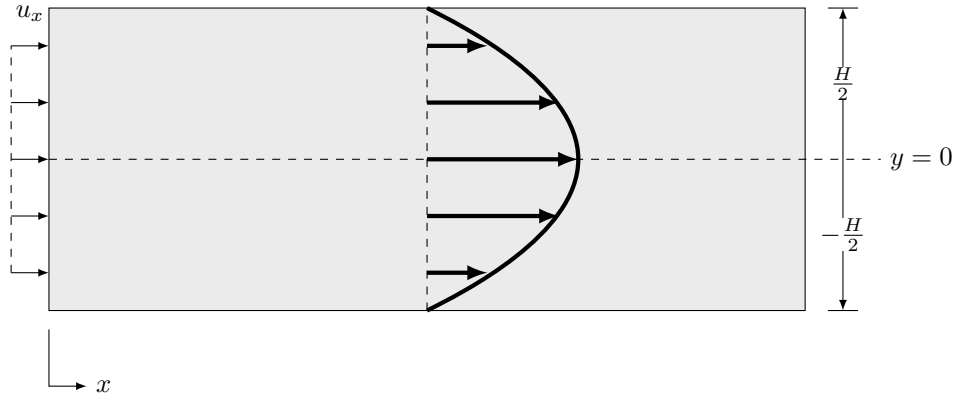


Figure 17. Schematic showing the 2D channel for fluid flow through two parallel plates used for analytical solution.

The coordinates are then transformed following [6, 12, 26, 27] as $\bar{y} = \frac{2y}{H}$ and $\bar{x} = \frac{2x/H}{PrRe'}$, where $Re' = \frac{2U_0H}{\nu}$ and Pr is the Prandtl number. The energy equation, Eqn. 29, becomes

$$\frac{3}{8} (1 - \bar{y}^2) \frac{\partial T}{\partial \bar{x}} = \frac{\partial^2 T}{\partial \bar{y}^2} + \left(\frac{1}{PrRe'} \right)^2 \frac{\partial^2 T}{\partial \bar{x}^2}. \quad (30)$$

The temperature field is then assumed to be of the form [26]

$$T(\bar{x}, \bar{y}) = Y(\bar{y}) e^{-\frac{8}{3}\lambda^2 \bar{x}}, \quad (31)$$

where T is the product of transverse coordinate $\bar{y}$, an exponentially decaying function in the flow direction $\bar{x}$, and λ which is a value that needs to be iteratively solved for. Important here is that this λ is not the same as λ_L in Eqn. 15, but the nomenclature is kept the same to be consistent with previous works. Substituting Eqn. 31 into Eqn. 30 gives

$$Y'' + \left[1 + \left(\frac{8\lambda}{3PrRe'} \right)^2 \right] \lambda^2 Y - \lambda^2 \bar{y}^2 Y = 0. \quad (32)$$

This is a Sturm-Liouville problem which can be expressed as a polynomial series [6]

$$Y(\bar{y}) = \sum_{i=0}^{\infty} b_i \bar{y}^{2i}. \quad (33)$$

After taking derivatives and applying boundary conditions, the following recursion relationship for the b_i coefficients arises:

$$b_0 = 1; b_1 = \frac{mb_0}{2}; b_i = \frac{mb_{i-1} + nb_{i-2}}{(2i)(2i-1)} \quad (i = 2, 3, 4, \dots), \quad (34)$$

where

$$m = - \left[1 + \left(\frac{8\lambda}{3PrRe'} \right)^2 \right] \lambda^2; n = \lambda^2. \quad (35)$$

To solve for λ , an expression for up to 30 coefficients of b_i is made using the sympy package [28] and the scipy package [29] is used to find the root by the Newton-Raphson method. Table 5 gives a summary of the eigenvalues obtained using this process at a Prandtl of 0.7 and varying Re' . There is no change in the coefficients from using an expression of 15, 20, and 30 coefficients of b_i , but the compute time increases drastically. For this work, 15 coefficients of b_i are used.

Table 5. Table of root values using 30, 25, 20, and 10 coefficients of b_i with a Prandtl number of 0.7 and varying Re' . Compute time for $Y(\bar{y})$ was performed on a single CPU core. Different digits are highlighted in red.

Coefficients					
Re'	30	25	20	15	10
50	1.66628076	1.66628076	1.66628076	1.66628076	1.666280 67
70	1.67366074	1.67366074	1.67366074	1.67366074	1.673660 65
100	1.67767423	1.67767423	1.67767423	1.67767423	1.677674 13
150	1.67984479	1.67984479	1.67984479	1.67984479	1.679844 70
200	1.68060910	1.68060910	1.68060910	1.68060910	1.680609 00
300	1.68115649	1.68115649	1.68115649	1.68115649	1.681156 39
500	1.68143725	1.68143725	1.68143725	1.68143725	1.681437 15
Time [min.]	235.15	17.75	1.59	0.15	0.02

References

- [1] S. V. Patankar, C. H. Liu, and E. M. Sparrow, “Fully developed flow and heat transfer in ducts having streamwise-periodic variations of cross-sectional area,” *Journal of Heat Transfer*, vol. 99, no. 2, pp. 180–186, 1977. [Online]. Available: <https://doi.org/10.1115/1.3450666>
- [2] O. Errico and E. Stalio, “Direct numerical simulation of turbulent forced convection in a wavy channel at low and order one prandtl number,” *International Journal of Thermal Sciences*, vol. 86, pp. 374–386, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.ijthermalsci.2014.07.021>
- [3] S. Harikrishnan and S. Tiwari, “Effect of skewness on flow and heat transfer characteristics of a wavy channel,” *International Journal of Heat and Mass Transfer*, vol. 120, pp. 956–969, 2018. [Online]. Available: <https://doi.org/10.1016/j.ijheatmasstransfer.2017.12.120>
- [4] A. Ciuffini, A. Scattina, F. Carena, M. Roberti, G. Toscano Rivalta, E. Chiavazzo, M. Fasano, and P. Asinari, “Multiscale computational fluid dynamics methodology for predicting thermal performance of compact heat exchangers,” *Journal of Heat Transfer*, vol. 138, no. 7, pp. 1–11, 2016. [Online]. Available: <https://doi.org/10.1115/1.4032980>
- [5] E. Stalio and M. Piller, “Direct numerical simulation of heat transfer in converging-diverging wavy channels,” *Journal of Heat Transfer*, vol. 129, no. 7, pp. 769–777, 2007. [Online]. Available: <https://doi.org/10.1115/1.2717235>
- [6] Z. Wang, H. Shang, and J. Zhang, “Lattice boltzmann simulations of heat transfer in fully developed periodic incompressible flows,” *Physical Review E*, vol. 95, no. 6, pp. 1–13, 2017. [Online]. Available: <https://doi.org/10.1103/PhysRevE.95.063309>
- [7] I. Ansys®, *Modeling Periodic Heat Transfer*. Ansys®, Inc., 2023, ch. 14.4.
- [8] OpenFOAM®, “OpenFOAM: User guide v2112,” 2023. [Online]. Available: <https://www.openfoam.com/>
- [9] A. Bjerg, K. Christoffersen, H. Sørensen, and J. Hærvig, “Flow structures and heat transfer in repeating arrangements of staggered rectangular winglet pairs by large eddy simulations: Effect of winglet height and longitudinal pitch distance,” *International Journal of Heat and Mass Transfer*, vol. 131, pp. 654–663, 2019. [Online]. Available: <https://doi.org/10.1016/j.ijheatmasstransfer.2018.11.015>
- [10] J. Hærvig, “cyclicTemperatureFoam-An OpenFOAM solver for cyclic heat transfer,” 2018. [Online]. Available: <https://doi.org/10.5281/zenodo.1254573>
- [11] S. Harikrishnan and S. Tiwari, *Simulation of Fully Developed Flow and Heat Transfer in Wavy Channels Using OpenFOAM*. Springer Singapore, 2020. [Online]. Available: https://doi.org/10.1007/978-981-15-0124-1_78
- [12] —, “Unsteady flow and heat transfer characteristics of primary and secondary corrugated channels,” *Journal of Heat Transfer*, vol. 142, no. 3, pp. 1–12, 2020. [Online]. Available: <https://doi.org/10.1115/1.4045751>
- [13] S. Harikrishnan, “Implementing streamwise periodic boundary condition in OpenFOAM,” *FOSSEE-CFD Case Study Project Report*, no. March, pp. 1–9, 2019. [Online]. Available: <https://doi.org/10.13140/RG.2.2.22674.09925>
- [14] J.-C. Han, *Analytical Heat Transfer*, 2nd ed. Boca Raton: Taylor and Francis Group, LLC, 2012.
- [15] V. Portyanikhin, “pyfluids,” 2023. [Online]. Available: <https://github.com/portyanikhin/PyFluids>
- [16] W. J. Minkowycz, E. M. Sparrow, and J. Y. Murthy, Eds., *Handbook of Numerical Heat Transfer*. John Wiley and Sons, Inc., 2006.
- [17] F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S. Lavine, *Incropera’s Principles of Heat and Mass Transfer, 8th Edition, Global Edition*. Singapore: John Wiley and Sons, 2017.
- [18] A. Bejan, *Convection Heat Transfer: Fourth Edition*, 4th ed. Hoboken, N.J.: Wily, 2013.
- [19] P. J. Roache, “Perspective: A method for uniform reporting of grid refinement studies,” *Journal of Fluid Engineering*, vol. 116, pp. 405–413, 1994. [Online]. Available: <https://doi.org/10.1115/1.2910291>
- [20] —, “Quantification of uncertainty in computational fluid dynamics,” *Annual Review of Fluid Mechanics*, vol. 29, pp. 123–160, 1997. [Online]. Available: <https://doi.org/10.1146/annurev.fluid.29.1.123>
- [21] H. M. Bahaidarah, N. K. Anand, and H. C. Chen, “Numerical study of heat and momentum transfer in channels with wavy walls,” *Numerical Heat Transfer; Part A: Applications*, vol. 47, no. 5, pp. 417–439, 2005. [Online]. Available: <https://doi.org/10.1080/10407780590891218>
- [22] A. G. Ramgadia and A. K. Saha, “Fully developed flow and heat transfer characteristics in a wavy passage: Effect of amplitude of waviness and reynolds number,” *International Journal of Heat and Mass Transfer*, vol. 55, no. 9-10, pp. 2494–2509, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.ijheatmasstransfer.2012.01.010>
- [23] G. Wang and S. P. Vanka, “Convective heat transfer in periodic wavy passages,” *International Journal of Heat and Mass Transfer*, vol. 38, no. 17, pp. 3219–3230, 1995. [Online]. Available: [https://doi.org/10.1016/0017-9310\(95\)00051-A](https://doi.org/10.1016/0017-9310(95)00051-A)
- [24] B. Ničeno and E. Nobile, “Numerical analysis of fluid flow and heat transfer in periodic wavy channels,” *International Journal of Heat and Fluid Flow*, vol. 22, no. 2, pp. 156–167, 2001. [Online]. Available: [https://doi.org/10.1016/S0142-727X\(01\)00074-1](https://doi.org/10.1016/S0142-727X(01)00074-1)
- [25] T. Nishimura, Y. Ohori, and Y. Kawamura, “Flow characteristics in a channel with symmetric wavy wall for steady flow,” *Journal of Chemical Engineering of Japan*, vol. 17, no. 5, pp. 466–471, 1984. [Online]. Available: <https://doi.org/10.1252/jcej.17.466>
- [26] G. M. Brown, “Heat or mass transfer in a fluid in laminar flow in a circular or flat conduit,” *American Institute of Chemical Engineers*, vol. 6, no. 2, pp. 179–183, 1960. [Online]. Available: <https://doi.org/10.1002/aic.690060204>
- [27] P. Li and J. Zhang, “The temperature decomposition method for periodic thermal flows with general wall conditions,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 74, no. 3, pp. 559–577, 2018. [Online]. Available: <https://doi.org/10.1080/10407790.2018.1538290>
- [28] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, v. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz, “SymPy: symbolic computing in python,” *PeerJ Computer Science*, vol. 3, p. e103, Jan. 2017. [Online]. Available: <https://doi.org/10.7717/peerj-cs.103>

- [29] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, I. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, “SciPy 1.0: Fundamental algorithms for scientific computing in python,” *Nature Methods*, vol. 17, pp. 261–272, 2020. [Online]. Available: <https://doi.org/10.1038/s41592-019-0686-2>