MATH3013 Final Project

OUYANG LAM CHEONG

May 20, 2024

Abstract

This project focuses on the development of mathematical models to simulate the thermal behavior when hot water is introduced into water environments, with a particular emphasis on home bathtubs.

In this project, two models are developed. The first model is based on Newton's cooling law and specific heat equations, providing a straightforward simulation to understand how water temperature changes over time. The second model incorporates spatial factors using the heat conduction equation with Neumann boundary conditions, allowing for a detailed representation of temperature distribution within the bathtub. Both models aim to provide insights of improving the process of temperature control by understanding the dynamics of heat transfer and distribution.

The results show that a slow, steady trickle of hot water is an effective strategy for maintaining a stable and comfortable water temperature in a bathtub. Sensitivity analyses reveal the significant influence of flow rate and inlet temperature on the model. The strengths and weaknesses of both models are discussed, and suggestions for further development are provided to enhance the accuracy of thermal simulations.

Contents

1 Introduction

1.1 Background

Maintaining a comfortable and consistent water temperature is crucial for various water environments, including home bathtubs, thermostatic swimming pools, and artificial hot springs. By precise temperature control, it is possible to ensure that the human body is always in a thermally comfortable water environment.

Home bathtubs provide a private space for relaxation and personal hygiene, requiring careful temperature management to maintain comfort. Thermostatic swimming pools, used for exercise, need to maintain a stable temperature to accommodate swimmers. Artificial hot springs, designed to mimic the natural geothermal experience, must balance the addition of hot water with user safety and comfort.

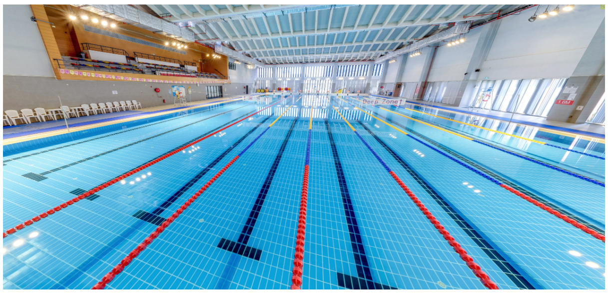

Figure 1: Swimming Pool in University of Macau

A common method to control water temperature in these environments is by adding hot water. This project aims to develop mathematical models to simulate the thermal behavior of water in these environments when hot water is introduced. By understanding the dynamics of heat transfer and distribution, the process of temperature control can be optimized and improved. In this project, we focus specifically on the scenario of a home bathtub.

1.2 Problem

The control of water temperature involves several challenges. These include the need to maintain a consistent temperature over time, manage heat loss to the environment, and ensure that the added hot water is evenly distributed.

In home bathtubs, users typically start with water at a desired temperature, but the water cools down as time passes. Then hot water is added to deal with the cooling. The challenge lies in determining the right amount and rate of hot water addition to maintain a steady temperature without causing discomfort or wasting water.

We have the following questions about the bathtub:

1. How can we maintain the temperature of bathtub water to keep comfort?
2. How does the temperature of bathtub water change when persistantly adding hot water?
3. What impact do different environmental of physical parameters have on the temperature change of bathtub water?
4. What is the spatial temperature distribution over time in the bathtub?

2 Assumptions

Two models are developed in this project. Here are the global assumptions of the two models. Local assumptions for each model will be made during the respective modelling process if needed.

- Ambient temperature is considered constant in space and time, 20°C.
- The initial temperature of the bathtub is spatially constant, 37°C.
- Any addition of water will cause an immediate overflow of water in the bathtub.
- The user is considered as an extension of the bathtub water since the person in the bathtub is in thermal equilibrium with the water.
- The water in the bathtub does not evaporate, that is, the amount of water stays the same.
- The density and specific heat of the water are constants in space and time. It's not necessary to consider their small variations.
- Heat transfer occurs by conduction and convection. There is no thermal radiation. This is a reasonable assumption at low temperatures.
- The bathtub is always in thermal equilibrium with the water, so its interface with the surrounding air can be described without needing to know the temperature distribution within the bathtub's material.
- The hot water added from faucet is stable.

3 Notations

Table 1: Notations

Symbol	Description
T/u	Temperature
T_∞/u_∞	Air temperature
t	Time
λ	Cooling constant
h	Heat transfer coefficient
S	Surface area
c	Specific heat capacity of water
m_{tub}	Mass of water in the bathtub
Q	Change in heat energy
Q_{in}	Heat from water flowing in
m_{in}	Mass of water flowing in
T_{in}/u_{in}	Temperature of water flowing in
Q_{out}	Heat from water draining out
m_{out}	Mass of water draining out
$\dot{m}$	Mass flow rate
h_t	Heat transfer coefficient through top surface
h_b	Heat transfer coefficient through bottom surface
h_w	Heat transfer coefficient through wall
L	Length of bathtub
W	Width of bathtub
H	Height of bathtub
α	Thermal diffusivity
$\tilde{Q}$	Heat generation
k	Thermal condutivity
ρ	Density of water
$\tilde{h}$	Heat transfer coefficient
$\hat{h}$	Heat transfer coefficient
J	Heat flux

4 Geometric consideration

In the following models, bathtub will be considered as a rectangular cuboid. That is, a three-dimensional shape composed of six rectangular faces, each face is a rectangle and each corner is a right angle. The upper rectangle is exposed to the air, and the remaining five sides are composed of the material of the bathtub. The x, y, and z directions correspond to its length, width, and height, respectively. The perspective view of the bathtub is shown in Figure 2.

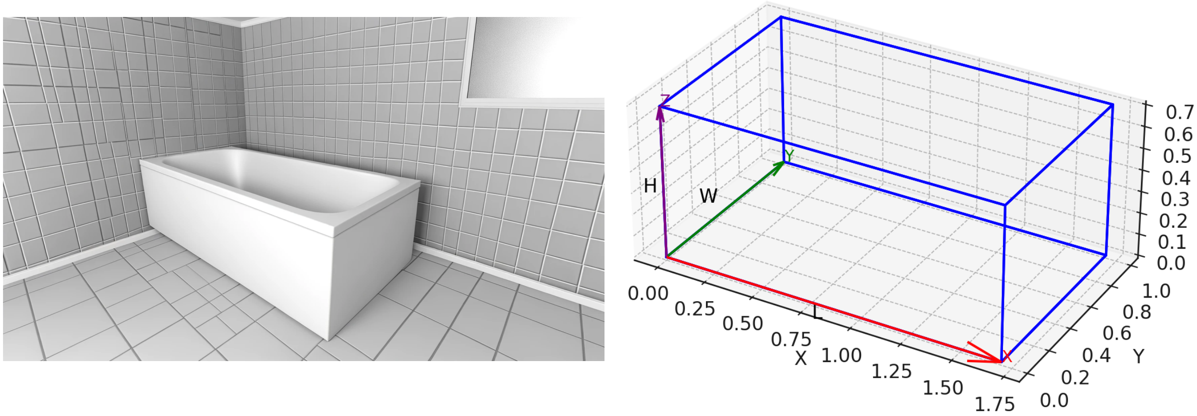

Figure 2: Bathtub and its perspective view

The faucet is placed at $x = 0$. Whenever water is added into the system, the same volume of water drains out from the system. L, W, and H represent its length, width, and height respectively. Here, the bathtub has dimensions of $1.7\text{m} \times 1.0\text{m} \times 0.7\text{m}$.

In general, it is a four-dimensional differential equation with generation and loss of heat. In order to reduce the complexity of the problem, in the PDE model we will assume that the flow and heat transfer of the water occur only in the x-direction, and that the temperature distribution in the yz-plane of a cross-section will be the same.

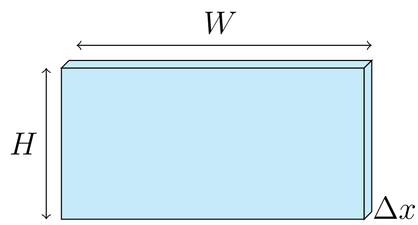

Figure 3: Cross section of bathtub

5 Model I

Based on Newton's cooling law and the specific heat equation, an ODE model is introduced here. For this model, the temperature of the water is assumed depending only on time. Spatial factors will be considered in another model.

In order to realize this assumption, the person in the bathtub must somehow make movements of their limbs so that the temperature of the water remains consistent throughout the bathtub.

5.1 Model Building

Cooling

- Newton's cooling law states that the rate of heat flow is proportional to the temperature difference between the surface and its surroundings. Also by the specific heat equation

$$Q = mc\Delta T \tag{1}$$

Equation of the cooling part is obtained below

$$\frac{dT}{dt} = -\frac{hS}{cm_{tub}}(T - T_\infty) \tag{2}$$

where T is the temperature of water. h is the Newton cooling coefficient. S is the surface area through which heat is lost. m_{tub} is the mass of water in the bathtub. T_∞ is the constant temperature of the surroundings.

Write $\lambda = \frac{hS}{cm_{tub}}$ and solve this equatiuon

$$\frac{dT}{T - T_\infty} = -\lambda dt$$
$$\ln|T - T_\infty| = -\lambda t + C$$
$$T(t) = T_\infty + C_1 e^{-\lambda t}$$
$$C_1 = T_0 - T_\infty$$
$$T(t) = T_\infty + (T_0 - T_\infty)e^{-\lambda t}$$

Plug in λ

$$T(t) = T_\infty + (T_0 - T_\infty)e^{-\frac{hS}{cm_{tub}}t} \tag{3}$$

Heating

- A person turns on the faucet. Hot water enters through the faucet, heating the water in the bathtub. The heat change due to this input is expressed as

$$Q_{in} = \frac{dm_{in}}{dt}cT_{in} \tag{4}$$

  where m_{in} is the mass of water entering per unit time, c is the specific heat capacity of water, T_{in} is the the temperature of the incoming hot water, and T is the current temperature of the bathtub water.

- The water in bathtub drains out when hot water enters.

$$Q_{out} = \frac{dm_{out}}{dt}cT \tag{5}$$

  where m_{out} is the mass of water draining out per unit time, T is the the temperature of the losing water, same as the temperature of the water in the tub.

Since the volumn of water flowing in equals the volumn of water flowing out. We write $\frac{dm}{dt} = \frac{dm_{in}}{dt} = \frac{dm_{out}}{dt}$. And therefore by subtraction

$$Q = Q_{in} - Q_{out} = \frac{dm}{dt}c(T_{in} - T) \tag{6}$$

By conservation of energy, we also have

$$Q = m_{tub}c\tilde{T}$$

where $\tilde{T}$ denotes the instantaneous rate of change in bathtub water temperature due to the inflow of hot water. Solve for $\tilde{T}$

$$\tilde{T} = \frac{dm}{dt}\frac{(T_{in} - T)}{m_{tub}}$$

We add the equations up, obtaining

$$\frac{dT}{dt} = -\frac{hS}{cm_{tub}}(T - T_{\infty}) + \frac{dm}{dt}\frac{(T_{in} - T)}{m_{tub}} \tag{7}$$

Solving this ode, we have

$$\frac{dT}{dt} = \left(\frac{-hS - \dot{m}c}{cm_{tub}}\right) T + \left(\frac{hST_{\infty} + \dot{m}cT_{in}}{cm_{tub}}\right)$$

where $\dot{m} = dm/dt$ is a constant. Also write

$$\eta = \left(\frac{-hS - \dot{m}c}{cm_{tub}}\right) \text{ and } \xi = \left(\frac{hST_{\infty} + \dot{m}cT_{in}}{cm_{tub}}\right)$$

$$\frac{dT}{dt} = \eta T + \xi$$

And mutiply both side by a integrating factor

$$\frac{d}{dt}\left(e^{-\eta t} T\right) = \xi e^{-\eta t}$$

$$T(t) = -\frac{\xi}{\eta} + Ce^{\eta t}$$

We notice that after turning on hot water, $T \to -\frac{\xi}{\eta}$ as $t \to \infty$, where $\xi > 0$ and $\eta < 0$.

$$\begin{cases} T(t) = -\dfrac{\xi}{\eta} + Ce^{\eta t}, \quad t \geq 0 \\ \xi = \left(\dfrac{hST_{\infty} + \dot{m}cT_{in}}{cm_{tub}}\right) \\ \eta = \left(\dfrac{-hS - \dot{m}c}{cm_{tub}}\right) \end{cases} \tag{8}$$

5.2 Parameters

- We consider a bathtub that is 1.7 meters long, 0.7 meters wide, and 0.7 meters deep.
- Mass of the water is calculated by $m = \rho V = 1000 \times (1.7 \times 1 \times 0.7) = 1190$kg
- Mass flow rate $\dot{m}$ varys from different faucets. It also can be controlled by the person. Here we assume $\dot{m} = 0.4$kg/s, which is a normal flow rate.
- The initial water temperature is set to be consistent with temperature of human body.
- Heat transfer coefficient and other parameters are listed in Table 2. The values can be modified depending on varing scenarios.
- The formula of hS is calculated by considering the six faces of the bathtub

$$hS = (h_t + h_b)LW + h_w(2HW + 2HL) = 216.16 \text{ W/K}$$

Table 2: Parameters and nominal values

Parameter	Description	Nominal Value
L	Bath Length	1.7 m
W	Bath Width	1 m
H	Bath Height	0.7 m
m	Mass of the water in the tub	1190 kg
$\dot{m}$	Mass Flow Rate	0.4 kg/s
T_{in}	Input Water Temperature	45°C
T_∞	Surroundings Temperature	20°C
T_0	Initial Water Temperature	37°C
ρ	Water Density	1000 kg/m^3
c	Specific Heat Capacity	4178 J/(kg·K)
h_t	Heat Transfer Coefficient of top	40 W/(m^2·K)
h_b	Heat Transfer Coefficient of bottom	16 W/(m^2·K)
h_w	Heat Transfer Coefficient of sides	32 W/(m^2·K)

5.3 Results

Equation 3 is implemented by using matlab. The temperature change within 30 minutes is ploted in figure 4.

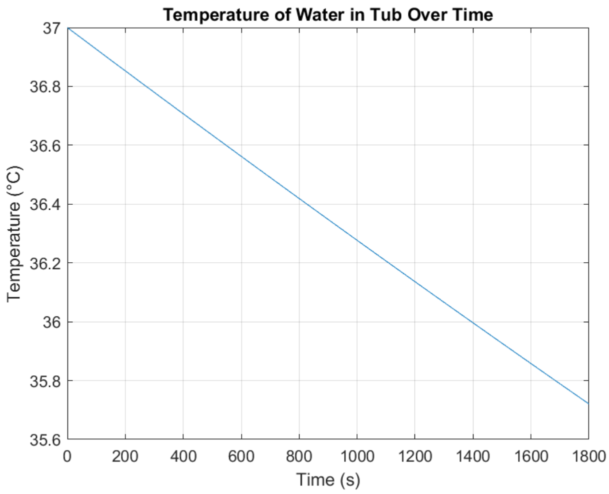

Figure 4: Temperature change within 30 minutes

When the temperature reachs significantly colder, the user in the bathtub turns on the hot water. At this time, Equation 3 can no longer be applied. The simulation will be proceeded by using Equation 8. The following figure shows the result.

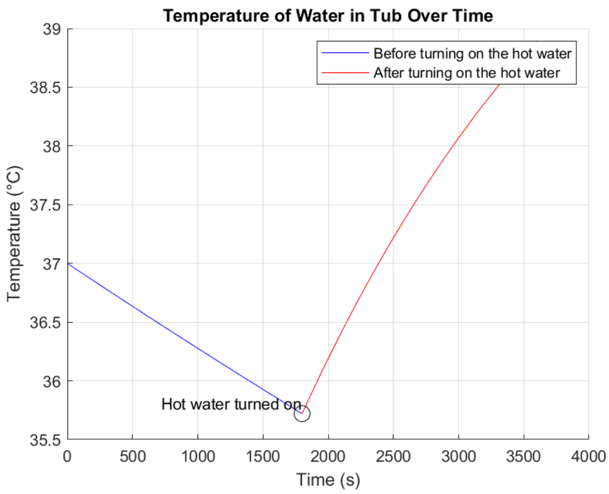

Figure 5: Temperature change within 1 hour

To maintain the water temperature around 37°C without wasting too much water, a strategy is to keep a slow, steady trickle of water flowing. This way, the person doesn't have to repeatedly turn the water on and off, ensuring an uninterrupted bathing experience.

We noticed that in Figure 6, turning on a 45°C water flow of around 0.1 kg/s can keep the water temperature in the bathtub stable at 37°C.

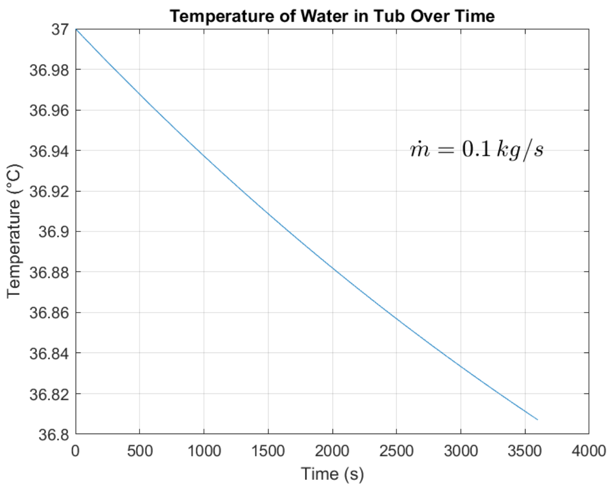

Figure 6: Temperature change with trickling hot water

Overall, the modeling results are consistent with our intuition. Ten minutes after turning on the hot water, the water in the bathtub returned to its initial temperature in Figure 5. Next, we naturally wonder about the effects on the model if variables such as water flow rate and ambient temperature are altered.

5.4 Sensitivity Analysis

The solution of the ordinary differential equation (7) is

$$T(t) = -\frac{\xi}{\eta} + Ce^{\eta t}$$

$T \to -\frac{\xi}{\eta}$ as $t \to \infty$, where $\xi > 0$ and $\eta < 0$. By considering different value of parameters, the steady water temperature in the bathtub will vary.

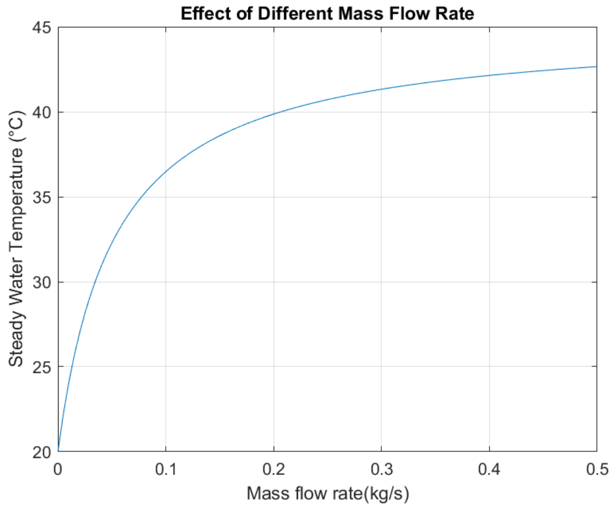

Figure 7: Sensitivity of steady temperature to flow rate

From the figure above, increasing rate of mass flow $\dot{m}$ causes water temperature gradually approach 45°C. It is clear that flow rate has a great influence on the model. Another parameter is illustated in the following.

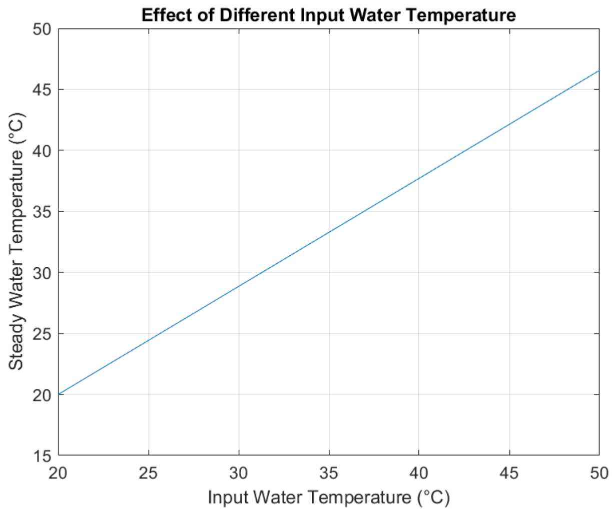

Figure 8: Sensitivity of steady temperature to inlet temperature

When 20°C water is continually added to the bathtub, the temperature of the water will remain at 20°C since the temperature is the same as the air temperature. When the temperature of added water T_{in} exceeds 20°C, there is a difference between the water temperature and the air temperature. Therefore, the final temperature decreases due to Newton's cooling law.

Finally, the sensitivity of air temperature is considered.

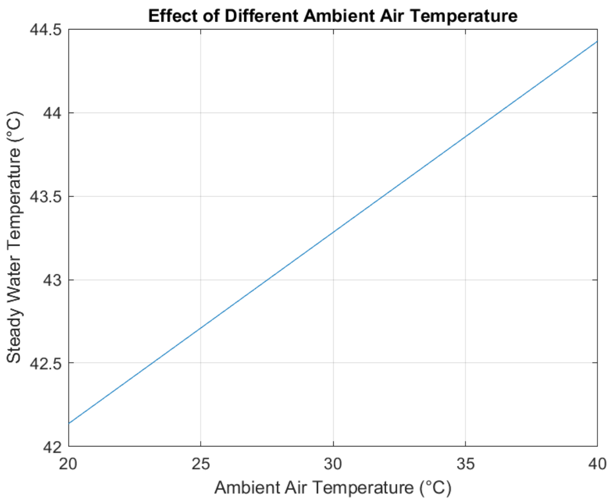

Figure 9: Sensitivity of steady temperature to air temperature

In the plot above, mass flow rate $\dot{m} = 0.4$ kg/s and input water temperature $T_{in} = 45$°C as in Table 2. It shows that the model is relatively less sensitive to the parameter of air temperature. However, ambient temperature still plays a role in determining the steady-state water temperature in the bathtub.

Model I provides a good insight of how the water temperature changes over time during a bath. However, considering the actual situation in reality where the faucet is installed on one side of the bathtub. The temperature distribution of water in a bathtub certainly varies with space when hot water is turned on. Therefore, another model will be introduced.

6 Model II

In this section, we will concentrate on the one-dimensional heat conduction equation which describes the transfer and diffusion of heat in a one-dimensional space. It is derived using the law of conservation of energy and Fourier's law of heat conduction. The standard form of the one-dimensional heat conduction equation is

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} \tag{9}$$

- $u(x,t)$ represents the temperature at position x and time t.
- α is the thermal diffusivity, defined as $\alpha = \frac{k}{\rho c}$, where k is the thermal conductivity of the material, ρ is the density of the material, and c is the specific heat capacity of the material.
- $\frac{\partial u}{\partial t}$ is the partial derivative of temperature with respect to time, representing the rate of change of temperature over time.
- $\frac{\partial^2 u}{\partial x^2}$ is the second partial derivative of temperature with respect to space, representing the curvature of the temperature distribution in space.

6.1 Model Building

A multi-dimensional heat conduction equation that includes a source term $\tilde{Q}$ is presented below

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u + \tilde{Q}(\vec{x}, t) \tag{10}$$

As stated in section 4, this equation is reduced to one spatial dimension such that the solution can be effectively obtained. Time dependent Neumann boundary conditions are suitable for describing the heat flux at the boundaries of a bathtub.

A general one dimension heat equation with Neumann broundary conditions is shown in Equation 11. This equation can be solved analytically, but we will only present numerical solution in this project.

$$\begin{aligned} \frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} + \tilde{Q}(x,t), \quad 0 < x < L, \quad t > 0, \quad \alpha \in \mathbb{R}, \\ u_x(x=0,t) = a(t), \quad t > 0, \\ u_x(x=L,t) = b(t), \quad t > 0, \\ u(x,t=0) = \phi(x), \quad 0 < x < L. \end{aligned} \tag{11}$$

Let's first consider a bathtub insulated on both ends, which is filled with 37°C water and cools over time. Similar to Equation 2, this pde only consider the cooling process. Also same as before, the greater the difference between the water temperature and the ambient temperature, the faster the water cools. Newton's cooling law is applied in the source term $\tilde{Q}(x,t)$.

$$\begin{aligned} \frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} + \tilde{Q}(x,t), \quad 0 < x < L, \quad t > 0, \quad \alpha \in \mathbb{R}, \\ u_x(x=0,t) = 0, \quad t > 0, \\ u_x(x=L,t) = 0, \quad t > 0, \\ u(x,t=0) = 37°\text{C}, \quad 0 < x < L, \\ \tilde{Q}(x,t) = \tilde{h}(u_\infty - u(x,t)), \quad 0 < x < L, \quad t > 0. \end{aligned} \tag{12}$$

where α is the thermal diffusivity, $\tilde{h}$ is the cooling coefficient, and $k = 0.6$ W/mK is the thermal conductivity of water. α and $\tilde{h}$ are calculated in this way

$$\alpha = \frac{k}{\rho c} = \frac{0.6}{1000 \times 4184} = 1.433 \times 10^{-7} \text{ m}^2/\text{s}$$

$$\tilde{h} = \frac{1}{\rho c}\left(\frac{2h_w}{W} + \frac{h_b}{H} + \frac{h_t}{H}\right)$$

The reason that $\tilde{h}$ is different from h in Model I is that PDE considers the local temperature changes within the water, requiring the heat transfer coefficient to be normalized to each small volume element.

For h, the temperature is uniformly distributed and we only need to consider the total heat loss throughout the body of water.

6.2 Numerical Scheme

We will use Explicit Finite Difference Method to solve Equation 12.

Discretization

Spatial step: $x_i = i\Delta x, \quad i = 0, 1, \ldots, N, \quad \Delta x = \frac{L}{N}$
Time step: $t_n = n\Delta t, \quad n = 0, 1, 2, \ldots$
where Δx and Δt are the spatial and temporal steps, respectively.

Finite Difference Scheme

For the time derivative, use the forward difference method

$$\left.\frac{\partial u}{\partial t}\right|_{i,n} \approx \frac{u_i^{n+1} - u_i^n}{\Delta t}$$

For the second derivative of the spatial derivative, use the central difference method

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_{i,n} \approx \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2}$$

Substitute the above discretizations into the PDE to obtain the explicit scheme

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + \tilde{h}(u_\infty - u_i^n)$$

$$u_i^{n+1} = u_i^n + \Delta t \left(\alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + \tilde{h}(u_\infty - u_i^n) \right)$$

Boundary condition

For the boundary conditions $u_x(x, 0) = u_x(L, t) = 0$, use the central difference method

$$\left.\frac{\partial u}{\partial x}\right|_{x=0} \approx \frac{u_1^n - u_0^n}{\Delta x} = 0 \quad \Rightarrow \quad u_1^n = u_0^n$$

$$\left.\frac{\partial u}{\partial x}\right|_{x=L} \approx \frac{u_N^n - u_{N-1}^n}{\Delta x} = 0 \quad \Rightarrow \quad u_N^n = u_{N-1}^n$$

Initial condition

$$u_i^0 = 37°\text{C}, \quad i = 0, 1, \ldots, N$$

Complete scheme

- $u_i^0 = 37°\text{C}, \quad i = 0, 1, \ldots, N$
- $u_1^n = u_0^n, \quad u_N^n = u_{N-1}^n, \quad n \geq 0$
- $u_i^{n+1} = u_i^n + \Delta t \left(\alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + \tilde{h}(u_\infty - u_i^n) \right), \quad i = 1, 2, \ldots, N-1, \quad n \geq 0$

Stability condition

$$\Delta t \leq \min\left(\frac{2}{\tilde{h}}, \frac{2}{\frac{4\alpha}{(\Delta x)^2} + \tilde{h}} \right)$$

6.3 Stability Analysis

The numerical scheme is

$$u_i^{n+1} = u_i^n + \Delta t \left(\alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + \tilde{h}(u_\infty - u_i^n) \right)$$

Subsitute with $u_i^n = G^n e^{ikx_i}$, where G is the amplification factor. Then after some calculation, we obtain

$$G^{n+1} = G^n \left(1 - \frac{2\alpha \Delta t}{(\Delta x)^2}(1 - \cos(k\Delta x)) - \tilde{h}\Delta t \right) + \Delta t \tilde{h} u_\infty$$

Since $\tilde{h}u_\infty$ is a constant term, we only focus on the stability conditions of the growth factor G.

$$G = 1 - \frac{2\alpha \Delta t}{(\Delta x)^2}(1 - \cos(k\Delta x)) - \tilde{h}\Delta t$$

Require $|G| \leq 1$

$$G = 1 - \frac{2\alpha \Delta t}{(\Delta x)^2}(1 - \cos(k\Delta x)) - \tilde{h}\Delta t$$

$$G_{max} = 1 - \tilde{h}\Delta t \quad \text{and} \quad G_{min} = 1 - \frac{4\alpha\Delta t}{(\Delta x)^2} - \tilde{h}\Delta t$$

Hence

$$\begin{cases} |1 - \tilde{h}\Delta t| \leq 1 \\ |1 - \dfrac{4\alpha\Delta t}{(\Delta x)^2} - \tilde{h}\Delta t| \leq 1 \end{cases}$$

Simplify the above expression

$$\begin{cases} 0 \leq \tilde{h}\Delta t \leq 2 \\ 0 \leq \Delta t \left(\dfrac{4\alpha}{(\Delta x)^2} + \tilde{h} \right) \leq 2 \end{cases}$$

Finally, the inequality is obtained

$$\Delta t \leq \min \left(\frac{2}{\tilde{h}}, \frac{2}{\frac{4\alpha}{(\Delta x)^2} + \tilde{h}} \right) \tag{13}$$

In the simulation, $\Delta x = 0.057$, $\Delta t = 1$, $\alpha = 1.433 \times 10^{-7}$ m^2/s, and $\tilde{h} = 3.4466 \times 10^{-6}$ Subsitute all the parameters into Equation 13, it is clear that the stability condition is satisfied. Hence we conclude our finite difference scheme is stable.

6.4 Implementation and Results

The figure of Equation 12 below illustates the temperature distribution in the bathtub. With two insulated ends, water temperature drops uniformly due to temperature differences.

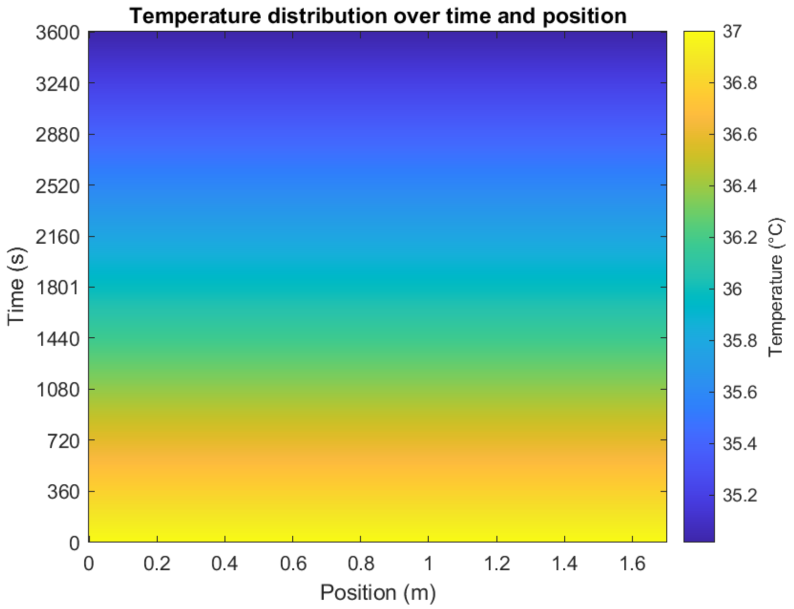

Figure 10: Temperature change within an hour

New boundary conditions ar $x = 0$ are considered in Equation 14. The boundary condition represents that hot water is added to the bathtub at $x = 0$, while the source term $\tilde{Q}$ represents that water is cooling.

$$
\begin{aligned}
\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} + \tilde{Q}(x,t), &\quad 0 < x < L, \quad t > 0, \quad \alpha \in \mathbb{R}, \\
u_x(x=0,t) = \hat{h}(u_{in} - u(0,t)), &\quad t > 0, \\
u_x(x=L,t) = 0, &\quad t > 0, \\
u(x,t=0) = 37°\text{C}, &\quad 0 < x < L, \\
\tilde{Q}(x,t) = \tilde{h}(u_\infty - u(x,t)), &\quad 0 < x < L, \quad t > 0.
\end{aligned}
\tag{14}
$$

α and $\tilde{h}$ are defined same as before. The coefficient $\hat{h}$ is defined as

$$\hat{h} = \frac{\dot{m}c}{kWH}$$

The reason we have such expression for $\hat{h}$ is as following. The heat flux J can be expressed as

$$J = -k\frac{\partial u}{\partial x}$$

By Equation 6

$$Q = \dot{m}c(u_{in} - u)$$

$$ku_x(x=0,t) = \frac{\dot{m}c}{WH}(u_{in} - u)$$

Rearranging the equation, obtaining

$$u_x(x=0,t) = \frac{\dot{m}c}{kWH}(u_{in} - u)$$

$\hat{h}$ is the heat transfer coefficient per unit length

$$\hat{h} = \frac{\dot{m}c}{kWH}$$

Now, we use Matlab's pdepe function to solve Equation 14. It handles parabolic and elliptic PDEs with time and space variables. The result of Equation 14 is shown in Figure 11. Due to low thermal diffusivity of water, the result is expected.

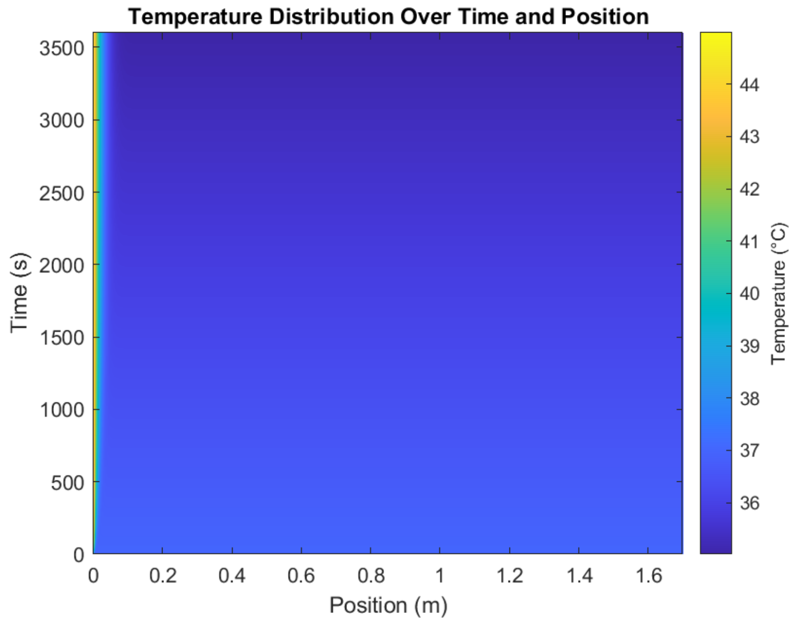

Figure 11: Temperature change with input hot water within an hour

As in the discussion of the problem in Section 1.1, in order to more clearly observe the variation of temperature with position, we increase the thermal diffusivity and observe Figure 12.

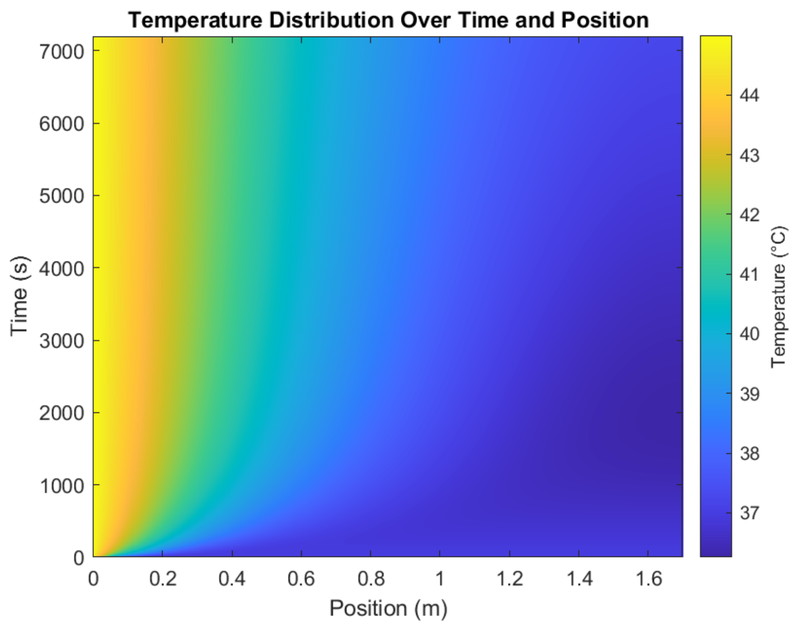

Figure 12: Temperature change with increased thermal diffusivity

- At position $x = 0$, the temperature can be seen rapidly rising to near 45°C (yellow). This is due to hot water entering the bathtub at position $x = 0$ and heating that area.

- Over time, heat propagates from the position $x = 0$ to the right, causing the temperature to gradually increase. This propagation demonstrates the effect of heat conduction.
- Near position $x = 1.7$, the temperature rise is slower because the heat takes time to propagate to farther locations.
- Still, the realistic scenario is ploted in Figure 11. The model is not accurate enough to capture all the significant nature. We will continue to discuss in section 8.

Figure 13 shows a scenario where there are faucets on both the left and right sides.

$$\begin{aligned}
\frac{\partial u}{\partial t} = \alpha\frac{\partial^2 u}{\partial x^2} + \tilde{Q}(x,t), &\quad 0 < x < L, \quad t > 0, \quad \alpha \in \mathbb{R}, \\
u_x(x=0,t) = \hat{h}[u_{in} - u(0,t)], &\quad t > 0, \\
u_x(x=L,t) = \hat{h}[u_{in} - u(L,t)], &\quad t > 0, \\
u(x,t=0) = 37°\mathrm{C}, &\quad 0 < x < L, \\
\tilde{Q}(x,t) = \tilde{h}(u_\infty - u(x,t)), &\quad 0 < x < L, \quad t > 0.
\end{aligned} \tag{15}$$

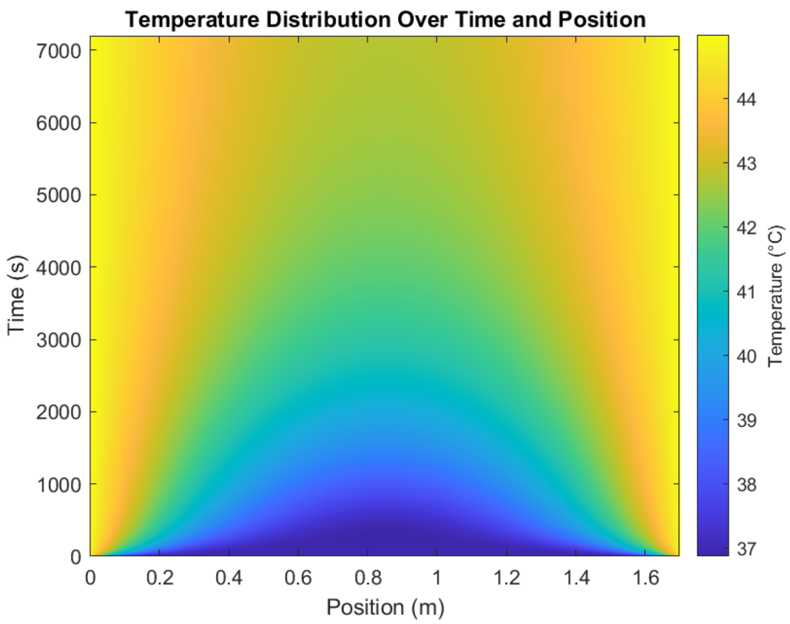

Figure 13: Temperature change with input hot water in two sides

In the context of bathtubs and swimming pools, it is possible that surfactants or disinfectants might be added. The thermal diffusivity α is affected when adding surfactants or disinfectants, but whether it increases or decreases depends on the nature and concentration of the added substances.

A sensitivity test of α to observe the impact of different values of α on temperature at $t = 2000$ s. The result is ploted in Figure 14.

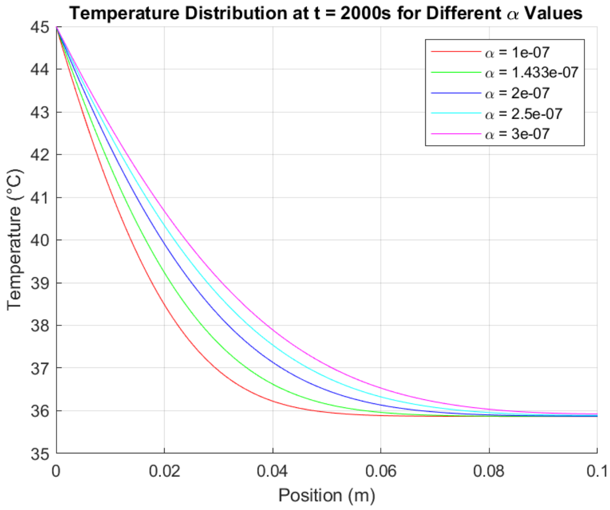

Figure 14: Sensitivity of temperature to thermal diffusivity

7 Strengths and Weaknesses

Strengths of Model I

- The straightforward simulation allows for quick calculations and results, which is beneficial for initial estimations or real-time applications.
- The model effectively captures the cooling behavior of water over time using Newton's cooling law, which is intuitive and aligns well with physical observations.
- The sensitivity analysis of various parameters such as flow rate, inlet temperature, and ambient temperature. This helps in understanding the influence of each parameter on the overall temperature dynamics.

Weaknesses of Model I

- It assumes a uniform temperature distribution throughout the bathtub, which is unrealistic, especially when hot water is added locally.
- By considering only time dependency and neglecting spatial factors, the model may not accurately reflect the actual thermal behavior in scenarios with significant spatial temperature gradients.
- The model's assumptions, such as immediate mixing and uniform temperature, limit its applicability to more complex systems where spatial change are significant.

Strengths of Model II

- It uses partial differential equations to consider the spatial distribution of temperature, providing a more detailed presentation of spatial temperature change within the bathtub.
- It offers a much comprehensive view of the thermal dynamics in the system.
- It can be adapted to other scenarios, such as different heat sources, varying boundary conditions.

Weaknesses of Model II

- The current formulation of the model does not include the advection term, which is crucial for accurately representing the movement of hot water within the bathtub. This omission means the model cannot fully capture the thermal dynamics caused by water flow.

- It increases the mathematical and computational complexity, requiring more advanced numerical methods and higher computational resources for solving the equations.

- Its accuracy is highly dependent on the precise values of parameters.

8 Model Improvement

It can be seen that the result is in fact unsatisfactory in Figure 11. Equation 14 considers the heat diffusion with a source term representing heat loss. However, this model assumes only diffusion and heat transfer at the boundaries.

An alternative model incorporates advection, representing the movement of water and heat due to the inflow of hot water. The advection-diffusion equation is

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} - \epsilon v \frac{\partial u}{\partial x} + \hat{Q}(x,t) \tag{16}$$

The appropriate values for the advection velocity v and the coefficient ϵ are needed to be determined. Modelling in this way, we can better simulate realistic scenarios and provide more accurate suggestion for maintaining water temperature.

In addition, we may consider the presence of user in the bathtub. That is to modify one of the assumption that user is regarded as an extension of the bathtub. This results in an extra heat transfer term with distinct heat transfer coefficient. It is highly possible to increase accuracy in both Models I and II.

9 Model Extension

The circulation and heating system of a swimming pool primarily involves the suction of water, filtration, heating, and returning the water back to the pool.

Water is drawn from the pool through various inlets. After filtration, the water passes through a heater. The heating system increases the water temperature to a confortable level for swimming. After being heated, the water is then pumped back into the pool through return inlets. These return inlets are strategically placed around the pool to ensure even distribution of the heated water.

The continuous cycle of suctioning water, filtering, heating, and returning it to the pool keeps the water in constant motion, which ensures that the pool remains clean and at a desirable temperature.

In a properly functioning swimming pool circulation system, the amount of water being added to the pool is equal to the amount of water being extracted, which is simlar to our model of bathtub.

However, swimming pools are significantly larger than bathtubs and have much complicated circulation systems with multiple inlets and outlets, while a bathtub only has a faucet and drain. Also, a swimming pool in operation will often be crowded with swimmers.

Therefore, extending the current bathtub model to swimming pools is challenging. It is necessary to consider more mechanism in order to effectively simulate the thermal behavior of a swimming pool.

10 Conclusion

In this project, two models are developed to simulate the thermal behavior of water in bathtubs when hot water is introduced. It might be possible to extend their application to other scenarios.

Model I based on Newton's cooling law and specific heat equations, provided a straightforward and quick simulation for understanding how temperature changes over time in a bathtub.

Model II incorporated spatial factors using the heat conduction equation with Neumann boundary conditions. This model allowed for a presentation of temperature distribution within the bathtub.

In summary, although the models cannot capture all the significant thermal dynamics, they provide valuable insights into the thermal behavior of water in bathtubs and similar environments. All answers listed in the section 1.2 were answered during the modeling process.

References

[1] B. Barnes and G.R. Fulford. *Mathematical Modelling with Case Studies Using Maple and MATLAB, Third Edition.* Textbooks in Mathematics. CRC Press, 2014.

[2] R.L. Burden and J.D. Faires. *Numerical Analysis.* Cengage Learning, 2010.

[3] Y.A. cCengel, J.M. Cimbala, M. Kanouglu, and R.H. Turner. *Fundamentals of Thermal-fluid Sciences.* McGraw-Hill Education, 2016.

[4] W.A. Strauss. *Partial Differential Equations An Introduction.* Wiley, 2007.

Matlab Code

```
% Define parameters
clc;
clear;
L = 1.7;            % Length of the tub
W = 1;              % Width of the tub
H = 0.7;            % Height of the tub
h_t = 40;           % Heat transfer coefficient (top)
h_b = 16;           % Heat transfer coefficient (bottom)
h_w = 32;           % Heat transfer coefficient (sides)
hS = (h_t + h_b) * L * W + h_w * (2 * H * W + 2 * H * L); %
    Total heat transfer coefficient
c = 4178;           % Specific heat capacity of water
m_tub = 1190;       % Mass of water in the tub
T_inf = 293.15;     % Ambient temperature
T0 = 310.15;        % Initial water temperature
T_in = 318.15;      % Input water temperature

% Calculate lambda
lambda = hS / (c * m_tub);

% Define the function f1 for temperature variation over time
f1 = @(t) T_inf + (T0 - T_inf) * exp(-lambda * t);

% Time range from 0 to 1800 seconds
t1 = 0:1:1800;

% Calculate the temperature at different time points
y1 = f1(t1) - 273.15;

% Plot the graph
figure;
plot(t1, y1);
xlabel('Time (s)');
ylabel('Temperature (Celsius)');
title('Temperature of Water in Tub Over Time');
grid on;

%%
% Temperature change before and after turning on hot water
m_dot = 0.4;
alpha = (-hS - m_dot * c) / (c * m_tub);
beta = (hS * T_inf + m_dot * c * T_in) / (c * m_tub);
Temp_switch = f1(1800);
C1 = (Temp_switch + beta / alpha);

% Update the definition of function f2
```

```matlab
f2_new = @(t) -beta / alpha + C1 * exp(alpha * (t - 1800));

% Time range
t2 = 1800:1:3600;

% Calculate the temperature
y2 = f2_new(t2) - 273.15;

% Plot the graph
figure;
hold on;
plot(t1, y1, 'b');
plot(t2, y2, 'r');
plot(1800, Temp_switch - 273.15, 'ko', 'MarkerSize', 10);
text(1800, Temp_switch - 273.15, ' Hot water turned on', '
    VerticalAlignment', 'bottom', 'HorizontalAlignment', 'right')
    ;
xlabel('Time (s)');
ylabel('Temperature (Celsius)');
title('Temperature of Water in Tub Over Time');
grid on;
legend('Before turning on the hot water', 'After turning on the
    hot water');
hold off;

%%
% Effect of mass flow rate on steady temperature
m_dot = linspace(0, 0.5, 100);
k = zeros(size(m_dot));
for i = 1:length(m_dot)
    alpha = (-hS - m_dot(i) * c) / (c * m_tub);
    beta = (hS * T_inf + m_dot(i) * c * T_in) / (c * m_tub);
    k(i) = -beta / alpha - 273.15;
end

% Plot the graph
figure;
plot(m_dot, k);
xlabel('Mass flow rate (kg/s)');
ylabel('Steady Water Temperature (Celsius)');
title('Effect of Different Mass Flow Rate');
grid on;

%%
% Effect of input water temperature on steady temperature
T_in_C = linspace(20, 50, 100);
T_in_K = T_in_C + 273.15;
k = zeros(size(T_in_C));
```

```
for i = 1:length(T_in_K)
    alpha = (-hS - m_dot * c) / (c * m_tub);
    beta = (hS * T_inf + m_dot * c * T_in_K(i)) / (c * m_tub);
    k(i) = -beta / alpha - 273.15;
end

% Plot the graph
figure;
plot(T_in_C, k);
xlabel('Input Water Temperature (Celsius)');
ylabel('Steady Water Temperature (Celsius)');
title('Effect of Different Input Water Temperature');
grid on;

%%
% Effect of ambient temperature on steady temperature
T_inf_C = linspace(20, 40, 100);
T_inf_K = T_inf_C + 273.15;
k = zeros(size(T_inf_C));
for i = 1:length(T_inf_K)
    beta = (hS * T_inf_K(i) + m_dot * c * T_in) / (c * m_tub);
    k(i) = -beta / alpha - 273.15;
end

% Plot the graph of k
figure;
plot(T_inf_C, k);
xlabel('Ambient Air Temperature (Celsius)');
ylabel('Steady Water Temperature (Celsius)');
title('Effect of Different Ambient Air Temperature');
grid on;

%%
% Flow rate 0.1
m_dot = 0.1;
alpha = (-hS - m_dot*c)/(c*m_tub);
beta = (hS*T_inf + m_dot*c*T_in)/(c*m_tub);
k = -beta/alpha - 273.15;
C = 310.15 + beta/alpha;
f3 = @(t) -beta/alpha + C*exp(alpha*t);
t = 0:1:3600;
y3 = f3(t) - 273.15;
plot(t, y3)
xlabel('Time (s)')
ylabel('Temperature (Celsius)')
title('Temperature of Water in Tub Over Time')
grid on
```

```
% Add text
text(2600, 36.94, '$$“dot–m” = 0.1 “, kg/s$$', 'Interpreter', '
    latex', 'FontSize', 15)

%%
clear;
clc;

% Parameters
L = 1.7; % Tub length
W = 1; % Tub width
H = 0.7; % Tub height
rho = 1000; % Water density
c = 4178; % Specific heat capacity of water
h_w = 32; % Heat transfer coefficient of the wall
h_b = 16; % Heat transfer coefficient of the bottom
h_t = 40; % Heat transfer coefficient of the top
u_inf = 20; % Ambient temperature
alpha = 0.6 / (rho * c); % Thermal diffusivity
Nx = 300; % Number of spatial steps
Nt = 3600; % Number of time steps
dx = L / Nx; % Spatial step size
dt = 1; % Time step size, s
x = linspace(0, L, Nx); % Spatial grid
t = linspace(0, Nt*dt, Nt); % Time grid

% Initialize temperature matrix
U = zeros(Nx, Nt);

% Initial conditions
U(:, 1) = 37; % Initial temperature distribution

% Calculate h_tilde
h_tilde = (2 * h_w / W + h_b / H + h_t / H) / (rho * c);


% Time evolution
for n = 1:Nt-1
    for i = 2:Nx-1
        U(i, n+1) = U(i, n) + alpha * dt / dx^2 * (U(i+1, n) -
            2*U(i, n) + U(i-1, n)) - h_tilde * dt * (U(i, n) - u_
            inf);
    end
    U(1, n+1) = U(2, n+1); % Neumann boundary condition, left
        boundary
    U(Nx, n+1) = U(Nx-1, n+1); % Neumann boundary condition,
        right boundary
end
```

```
% Plot the graph
figure;
imagesc(x, t, U.');
colorbarHandle = colorbar;
ylabel(colorbarHandle, 'Temperature (Celsius)')
xlabel('Position (m)');
ylabel('Time (s)');
title('Temperature distribution over time and position');

% Set y-axis ticks and labels
yticks = linspace(1, Nt, 11); % Select 11 tick points
yticks = round(yticks); % Ensure yticks are integers
yticklabels = arrayfun(@(y) sprintf('%.0f', t(y)), yticks, '
   UniformOutput', false);
set(gca, 'YTick', yticks, 'YTickLabel', yticklabels);
set(gca, 'YDir', 'normal');

clc;
clear;
bathtubPDE

function bathtubPDE
    % Parameters
    L = 1.7; % Length of the bathtub
    u_in = 45; % Input water temperature
    u_inf = 20; % Ambient temperature
    W = 1; H = 0.7; % Dimensions of the bathtub
    mdot = 0.4; k = 0.6; c = 4178; % Physical parameters
    h_t = 40; h_b = 16; h_w = 32; % Heat transfer coefficients
    alpha = 1.433e-7; % Thermal diffusivity
    h_hat = mdot * c / (k * W * H);
    h_tilde = 1 / (1000 * c) * (2 * h_w / W + h_b / H + h_t / H)
       ;

    m = 0;
    xmesh = linspace(0, L, 600); % Spatial grid
    tspan = linspace(0, 3600, 3600); % Time grid
    options = odeset('RelTol',1e-4,'AbsTol',1e-7);

    % Solving the PDE for the first part
    sol = pdepe(m, @pdefun, @icfun, @bcfun, xmesh, tspan,
       options);

    % Plotting the results for the first part
    figure;
    imagesc(xmesh, tspan, sol);
    title('Temperature Distribution Over Time and Position')
```

```
xlabel('Position (m)')
ylabel('Time (s)')
colorbarHandle = colorbar;
ylabel(colorbarHandle, 'Temperature (Celsius)')
axis xy;

% Parameters for the second part
L = 0.1; % Length of the bathtub
xmesh = linspace(0, L, 600);
tspan = linspace(0, 2000, 2000);
alpha_values = [1e-7, 1.433e-7, 2e-7, 2.5e-7, 3e-7];
colors = ['r', 'g', 'b', 'c', 'm'];

figure;
hold on;

for i = 1:length(alpha_values)
    alpha = alpha_values(i);
    % Solving the PDE for different alpha values
    sol = pdepe(m, @pdefun, @icfun, @bcfun, xmesh, tspan,
        options);

    % Plotting the results at t = 2000s
    plot(xmesh, sol(end, :), colors(i), 'DisplayName', ['“
        alpha = ', num2str(alpha)]);
end

title('Temperature Distribution at t = 2000s for Different “
    alpha Values')
xlabel('Position (m)')
ylabel('Temperature (Celsius)')
legend('show')
grid on
hold off;

function [c, f, s] = pdefun(x, t, u, DuDx)
    c = 1;
    f = alpha * DuDx;
    s = h_tilde * (u_inf - u);
end

function u0 = icfun(x)
    u0 = 37; % Initial temperature
end

function [pl, ql, pr, qr] = bcfun(xl, ul, xr, ur, t)
    pl = h_hat * (u_in - ul);
    ql = 1;
```

```
        pr = 0;
        qr = 1;
    end
end
```