NE 860 Nuclear Reactor Thermal-Hydraulics Computer Project #1

Daniel Franken
Graduate Student
Department of Mechanical and Nuclear Engineering
Kansas State University
Manhattan, Kansas 66502
Email: dfranken@ksu.edu

The following report discusses the results and conclusions developed during the ME 860 Nuclear Reactor Thermal-Hydraulics Computer Project #1.

Nomenclature

T Temperature [°C]
T_w Temperature at Wall[°C]
x Distance from 0 in the domain (0,L) [m]
k Thermal Conductivity [W/m-°C]
$q_\circ^{'''}$ Average Volumetric Heat Generation [W/m^3]
$q_1^{'''}$ Peak Volumetric Heat Generation [W/m^3]
L Plate Thickness [m]
C_n Unknown Coefficient
α Thermal Diffusivity [m^2/s]
λ_n Eigenvalues
Δx Incremental step in space [m]
Δt Incremental step in time [s]

1 Introduction

The computer project #1 focuses on 1D-Heat Conduction with and without heat generation in both steady state and transient conditions. In order to verify the accuracy of the results the solutions to the 1D-Heat conduction equations are solved both analytical and numerical.

2 Problem Statement

A flat plate fuel element of thickness 1 cm is composed of fresh UO_2 fuel. Assume thermal conductivity, density and specific heat to be constant throughout the domain during any transient (use literature values defined at 500°C). Bulk coolant temperature is assumed to be at 300°C which is cooling the fuel element on both sides. Volumetric heat generation rate in the fuel element varies as a sinusoidal function along the thickness of the fuel element with peak value at the center and zero at the boundaries. Average value of volumetric heat generation in the fuel during normal operation is 300 MW/m^3.

3 Part A Problem Statement

With the coolant heat transfer coefficient of 25000 W/m^2-°C, obtain and plot steady state temperature distribution in the fuel element as a function of thickness.

3.1 Part A Solution

The following is 1D steady state heat conduction equation.

$$k\frac{d^2T}{dx^2} + q''' = 0 \quad (1)$$

The problem statement states the average value of the volumetric heat generation in the fuel during normal operation is 300 MW/m^3. Additionally, it is also stated that the volumetric heat generation rate varies as a sinusoidal function along the thickness of the element, with a peak in the center and zero at the boundaries. Therefore, in order to account for these variations in heat generation along the length of the fuel the following equation was used:

$$q''' = \frac{q_o^{'''}}{\frac{2}{\pi}} sin(\frac{\pi x}{L}) \quad (2)$$

Since the problem gives the average volumetric heat generation it is important to be able to equate that to a sinusoidal function with the same average over the length of the domain. In order to have a sinusoidal function which has the equivalent average heat generation the function must be averaged over distance.The $\frac{\pi}{2}$ term is what is generated to convert the average volumetric heat generation to the peak volumetric heat generation value in equation 2. Note that peak volumetric heat generation $\frac{q_o'''}{\frac{2}{\pi}}$ has been replaced with q_1''' to reduce the number of terms.

The domain has been defined from 0 to L, with L being the thickness of the plate. This domain definition allows the use of the following sinusoidal function $sin(\frac{\pi x}{L})$. This sine function creates the condition where heat generation peaks at the center and goes to 0 at the bounds.

With the average volumetric heat generation converted into a sine function, equation 2 can now be inserted back into equation 1, this leads to the following equation:

$$\frac{d^2T}{dx^2} = -\frac{q_1'''}{k} sin(\frac{\pi x}{L}) \tag{3}$$

Integrating twice the equation then becomes:

$$T = \frac{L^2 q_1'''}{\pi^2 k} sin(\frac{\pi x}{L}) + C_1 x + C_2 \tag{4}$$

Both C_1 and C_2 are coefficients which are still to be determined. The boundary conditions used to solve this equation were that $T(x=0) = T(x=L) = T_w$, where T_w is used to denote the wall temperature at the boundary. Using theses boundaries the equation then simplifies to:

$$T = \frac{L^2 q_1'''}{\pi^2 k} sin(\frac{\pi x}{L}) + T_w \tag{5}$$

In order to solve for T_w the convective heat transfer equation is used.

$$q_o''' L = h(T_w - T_\infty) \tag{6}$$

T_∞ is the bulk fluid temperature of the coolant which is 300°C. Solving for T_w, it is seen that $T_w = 420°C$. For the thermal conductivity the following reference value of 4.5 $\frac{W}{m-K}$ was used [1]. Now that all of the unknowns are solved for, the temperature distribution can be computed and plotted. The plotted steady state temperature distribution can be seen in Figure 1.

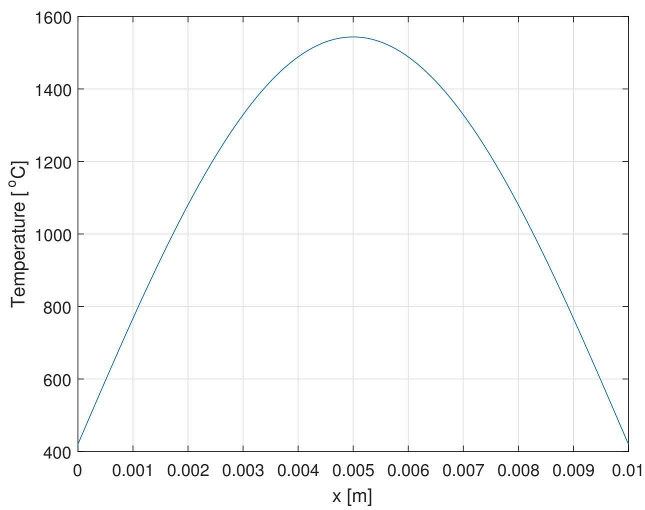

Fig. 1. Steady State Temperature Profile

4 Part B Problem Statement

After one year of operation at steady-state, due to a sudden transient i.e. tripping of coolant pumps the reactor is tripped. Due to which, the heat removal from one side of fuel element goes to zero. The other side is maintained at a constant temperature of 300^{o}C with the help of emergency injection.

Using the methods listed below, obtain temperature distribution in fuel element after 5 sec. of trip under two scenarios (i) No decay heat generation; (ii) Decay heat generation is equivalent to a constant value of 2% of full power. Plot the temperature distribution in the domain at 5 sec under both scenarios. Plot the maximum temperature versus time for both cases.

a) Analytical method (Separation of variables)
b) Numerical method (Finite difference)

4.1 Part B Solution - No Decay Heat Generation

4.1.1 Analytical Method (Separation of Variables)

The following is the 1D heat conduction with no heat generation equation:

$$\frac{\partial^2 T}{\partial^2 x} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{7}$$

The boundary conditions to this problem are:

$$B.C.\#1 : T(x=L) = T_w = 300°C \tag{8}$$

$$B.C.\#2 : \frac{dT}{dx}(x=0) = 0 \tag{9}$$

Since the transient begins after the steady state condition, the initial condition is just the solution of the steady state temperature profile solved in Part A.

$$I.C. : T(x,t=0) = \frac{L^2 q_1'''}{\pi^2 k} sin(\frac{\pi x}{L}) + T_w \tag{10}$$

In order to solve this problem a separation of variables must be performed in order to decouple the problem in the spatial and temporal domain. However, before that can be done, in order to make derivation simpler the temperature will be replaced with a temperature variable θ, being defined in equation 11. The new heat equation then becomes:

$$\theta = T - T_w \tag{11}$$

$$\frac{\partial^2\theta}{\partial^2 x} = \frac{1}{\alpha}\frac{\partial\theta}{\partial t} \tag{12}$$

Next, in order to separate the variables let $\theta(x,t) = X(x)\Gamma(t)$. Replace the new definition of θ into equation 12 to get:

$$\frac{1}{\alpha\Gamma(t)}\frac{d\Gamma(t)}{dt} = \frac{1}{X(x)}\frac{d^2X(x)}{dx^2} = -\lambda^2 \tag{13}$$

Solving for $\Gamma(t)$ it can be seen that:

$$\Gamma_n(t) = D_n e^{-\alpha\lambda_n^2 t} \tag{14}$$

The following is the general solution of $X(x)$:

$$X(x) = C_1 sin(\lambda x) + C_2 cos(\lambda x) \tag{15}$$

Since X(x) is for the spatial component of θ it is necessary to convert the boundary and initial conditions to account for the change from T to θ. The new boundary conditions then become:

$$B.C.\#1 : X(x = L) = 0 \tag{16}$$

$$B.C.\#2 : \frac{dX}{dx}(x = 0) = 0 \tag{17}$$

The initial condition is also changed to:

$$I.C. : \theta(x,t = 0) = \frac{L^2 q_1'''}{\pi^2 k} sin(\frac{\pi x}{L}) + T_{w,a} - T_{w,b} \tag{18}$$

It should be noted that in equation 18 that there are two wall temperatures listed in the equation. This is because the steady state solution had a different wall temperature than the new transient solution. In Part A the $T_w = T_{w,a} = 420°C$, while in Part B it is $T_w = T_{w,b} = 300°C$.

Using the boundary conditions in equations 16 & 17 on equation 15 it can be seen that $C_1 = 0$, while C_2 is a function of the infinite sum of eigenvalues. This leads to a solution for $\lambda_n = \frac{(2n-1)\pi}{2L}$. Therefore, by multiplying the solutions of $X(x)$ and $\Gamma(t)$ together θ can now be solved for.

$$\theta(x,t) = X(x)\Gamma(t) = \sum_{n=1}^{\infty} C_n cos(\lambda_n x) e^{-\alpha\lambda_n^2 t} \tag{19}$$

Before the final solution can be found, the unknown coefficients of the eigenfunction expansion need to be solved for. By using the orthogonality rule the coefficients can be solved for in terms of the fraction of two integrals.

$$C_n = \frac{\int_0^L f(x) cos(\lambda_n x) dx}{\int_0^L cos^2(\lambda_n x) dx} \tag{20}$$

The $f(x)$ listed in equation 20 is the initial condition of the problem listed in equation 18. Using equation 20 the coefficients to the eigenvalue expansion can be solved then those coefficients can be used to solve for θ. Once θ has been solved for, it can be substituted back into equation 11 in order solve for T analytically, therefore $T(x,t) = \theta(x,t) + T_{w,b}$.

4.1.2 Numerical Method (Finite Difference)

In order to solve the problem numerically the differential equation must be discretized. Starting with equation 7, the spatial double derivative is discretized in a second order central scheme, while the temporal derivative is take as a forward derivative. This scheme is also referred to as Forward in Time and Central in Space (FTCS). The discretized equation developed can be seen in equation 21. The solution is found by solving for T_i^{n+1}. Δx and Δt are small changes in x and t, respectively.

$$\frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2} = \frac{1}{\alpha}\frac{T_i^{n+1} - T_i^n}{\Delta t} \tag{21}$$

4.1.3 Solutions to the Analytical and Numerical Methods

The solutions to the numerical and analytical equations are used to create Figure 2 and 3. It can been seen that the figures are in fairly good agreement at the 5 second mark. The slight discrepancy between values can be attributed to the spatial and temporal resolution of the numerical results. Increasing the resolution of the numerical study should bring the solution closer to the analytical results.

4.2 Part B Solution - Decay Heat Generation

Because this problem has heat generation and a transient component all of the terms in the 1D heat conduction equation are used.

$$\frac{\partial^2 T}{\partial^2 x} + \frac{q'''}{k} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{22}$$

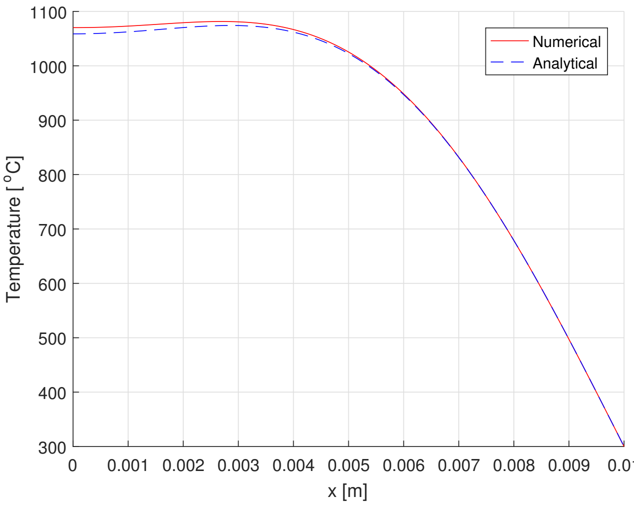

Fig. 2. Temperature Profile after 5 Seconds with Analytical and Numerical Methods

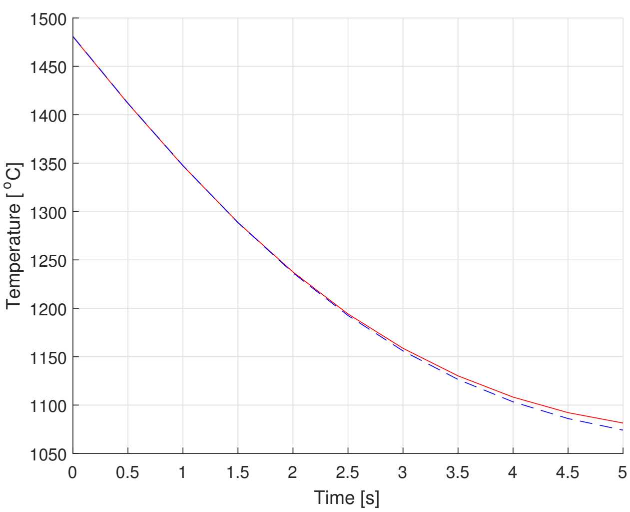

Fig. 3. Max. Temps. of Analytical and Numerical Methods with Time

To solve this problem the steady state and transient solutions must be split. There is an assumed steady state and transient temperature distribution contributing to the entire solution. Therefore, $T = T_{ss} + T_{tran}$.

4.2.1 Analytical Method (Separation of Variables)

The steady state solution is solved much in the same way as was done in Part A. Since the steady state solution does not take into consideration the transient terms of the solution, those terms are set equal to zero. After integrating equation 1 to get equation 3, the new boundary conditions must be applied.

The boundary conditions created in equation 16 & 17 can be applied to the steady state solution in equation 3 to acquire the following equation:

$$T_{ss}(x) = T_{w,b} + \frac{L^2 q_1^{'''}}{\pi^2 k} sin(\frac{\pi x}{L}) + \frac{L q_1^{'''}}{\pi k}(L - x) \quad (23)$$

Now that the steady state solution has been solve it is time to solve for the transient component of the solution. In order to simplify the solution the temperature will be alternated by replacing T with θ as what was done in equation 11. The transient solution derivation is performed in the same way as was done in Part B with no decay heat generation. This leads to the transient solution which looks similar to equation 19.

$$T_{tran}(x,t) = X(x)\Gamma(t) = \sum_{n=1}^{\infty} C_n cos(\lambda_n x) e^{-\alpha \lambda_n^2 t} \quad (24)$$

Adding equation 23 & 24 together the final solution can be found as $T = T_{ss} + T_{tran}$. It should be noted that $f(x)$ contained the steady state solution from Part A subtracted by $T_{w,b}$ for the no heat generation case, while for the heat generation case was subtracted by the steady state solution in Part B. The subtraction referenced here can be seen in Appendix A.2 & A.3.

4.2.2 Numerical Method (Finite Difference)

As mentioned in section 4.1.2 the solution for the numerical method is the same for the decay heat generation case. The only difference is that there is an added term used to describe the heat generation throughout the plate.

$$\frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2} + \frac{q_1^{'''} sin(\frac{\pi(\Delta x) i}{L})}{k} = \frac{1}{\alpha} \frac{T_i^{n+1} - T_i^n}{\Delta t} \quad (25)$$

Again, solving for T_i^{n+1} a time marching solution can be produced. This scheme is also FTCS even though there is an additional heat generation term.

4.2.3 Solutions to the Analytical and Numerical Methods

The analytical and numerical solutions are shown in Figure 4 & 5. Figure 4 shows the temperature distribution in space with a 2% heat generation rate, while Figure 5 shows the max temperature in both solution techniques. The values for density and specific heat capacity were taken from the International Atomic Energy Agency materials report [2].

5 Part C Problem Statement

Compare analytical and numerical methods by plotting the results (Temperature vs. thickness) at different times (e.g. 2 sec, 20 sec, 100 sec, 1000 sec). Show the convergence studies for both methods and order of accuracy for numerical method.

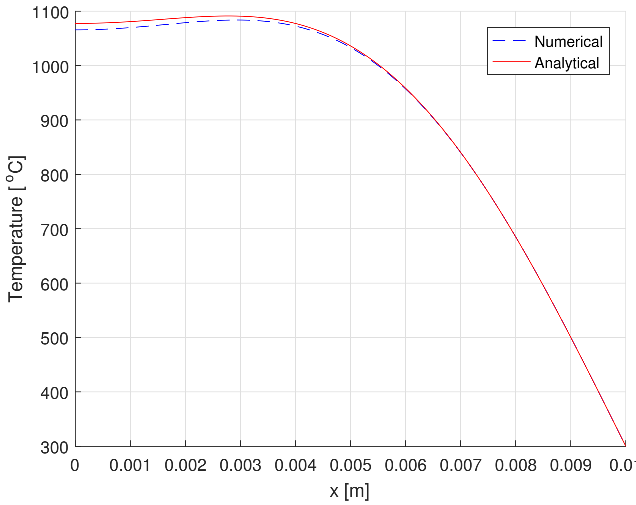

Fig. 4. Temperature Profile after 5 Seconds with Analytical and Numerical Methods

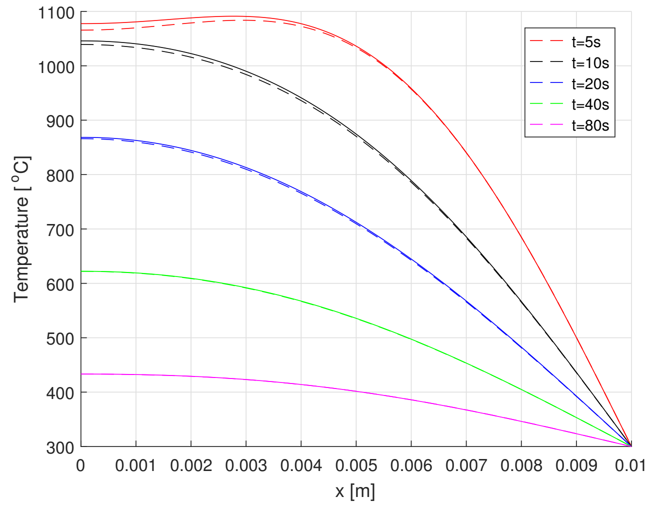

Fig. 6. Analytical vs Numerical at Different Times (Analytical = Dashed line, Numerical = Solid)

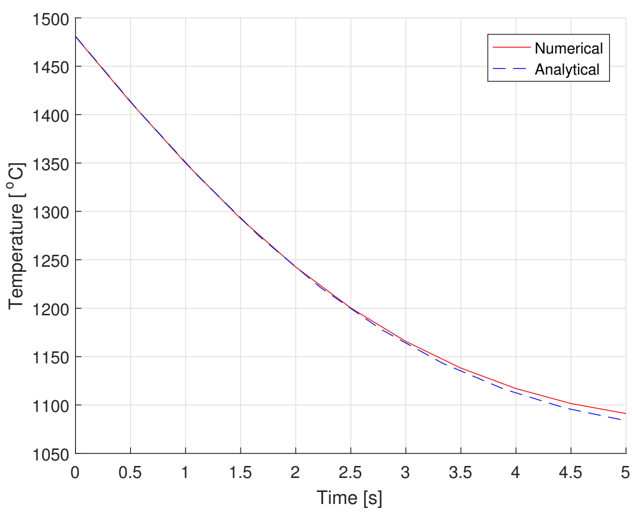

Fig. 5. Max. Temps. of Analytical and Numerical Methods with Time

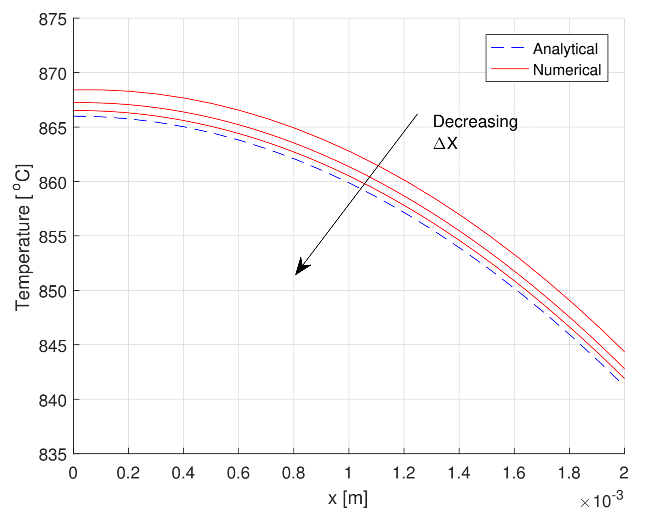

Fig. 7. Analytical vs Numerical results at t = 20s with different mesh sizes

5.1 Results at Different Times

It can be see from Figure 6 that the analytical and numerical results begin to converge as time grows. At t = 20s the results appear to be almost identical. It is expected that the error shown in Figure 6 may be reduced by increasing the number of mesh points and decreasing the time step. It should be noted that at t = 5s that the results do differ more so than the other times, this is thought to be due to the resolution of the numerical results.

Additional error may be reduce in the analytical solution by increasing the number of terms included in the solution. Figure 6 was created with a mesh with 100 points and a time step of 0.0001s. Additionally, the analytical solution used 40 terms in the eigenfunction expansion equation in the solution for T.

Shown in Figure 7 & 8 are the results of a numerical study performed to look at the effect of mesh size and time step on the numerical results. The time step was fixed at 0.0001 seconds for each mesh size in Figure 7. Figure 7 starts out with 100 mesh points and increases to 200, then 400 mesh points. It can be seen that as the number of mesh points increases the solution tends to converge towards the analytical solution. In Figure 8 the time step is decreased from 0.001s to 0.0001s and finally 0.00001s. It can be seen that the decrease in the time step does little to increase to accuracy of the solution. In fact, as the time step is decreased the solution tends to move away from the analytical solution, albeit in a very small way. The numerical method had a second order accuracy in space, and a first order accuracy in time.

Figure 9 shows the results of a convergence study performed on the analytical solution. The top line in Figure 9 is made with just one C_n term. The bottom line depicts higher number of terms in the analytical solution. The reason there are only two lines depicted is that the solution very

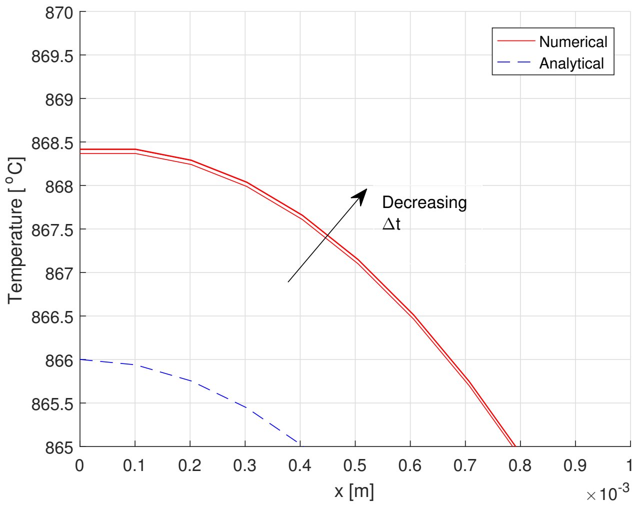

Fig. 8. Analytical vs Numerical results at different times with N = 100pts

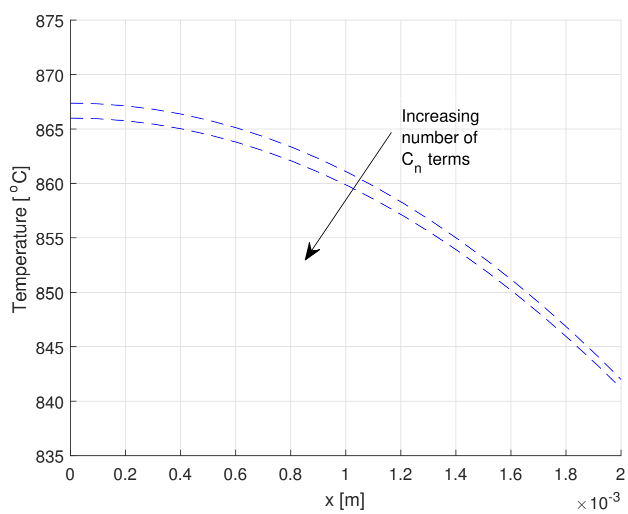

Fig. 9. Analytical results at t = 20s, with different number of C_n terms

quickly converged. Higher number of terms were tested but the changes were so small that it is difficult to see in the plot. The bottom lines are composed of a number of terms starting with 10 going to 1000, but since convergence is so quick the lines overlap each other.

6 Conclusions

Starting with Part A the steady state solution to the 1D heat conduction was derived. It was seen by applying appropriate boundary conditions the 1D heat equation could solved in a manner which described the temperature profile throughout the domain. In Part B it was seen how a transient solution could be derived using the eigenfunction expansion method. Additionally, it was also seen how a heat generation term can effect the equation for transient solution. Part C showed that given a small enough ΔX and time step Δt the numerical results generated can closely match the analytical results.

7 Citing References

References

[1] International Atomic Energy Agency, 1966. “Thermal Conductivity of Uranium Dioxide”. p. 80.

[2] Iaea, 2008. “Thermophysical Properties of Materials for Nuclear Engineering: A Tutorial and Collection of Data”. *Atomic Energy*, p. 200.

A Appendix A: Code

A.1 Part A Code

```
L = 0.01; %Length of the domain [m]
% Initialize domain
x = linspace(0,L,100);
% Plot initial condition
Tw = 420; %Temperature at the wall [Celsius]
k = 4.5; %Thermal Conductivity [W/mK]
q = 300E6/(2/pi); %Converts Average Heat Gen. To Peak Heat Gen
T = Tw + (((L^2)*q*sin((pi*x)/(L))/((pi^2)*k))); %From 0 to L
plot(x,T)
grid on
xlabel('x [m]')
ylabel('Temperature [^{o}C]')
```

A.2 Part B Numerical Method Code

```
%%%%%USE T FROM PART A AS INITIAL CONDITION%%%%%
k = 4.5; %Thermal Conductivity [W/mK]
rho = 11000; %Density of UO2 [kg/m^3]
c = 300; %Specific Heat Capcity of UO2 [J/kgK]
alpha = k/(rho*c); %Thermal Diffusivity [m^2/s]
q = (300E6/(2/pi))*0.02; %2% Peaking Power [W]
h = L/(N-1); %Delta x
dt = 0.001; %Delta t
% Initialize domain
x = linspace(0,L,N);
% Calculate heat at specified times
% t = [5,10,20,40,80]; %Used to calculate at multiple times
t = 20; %Used to calculate at one time
% t = linspace(0,5,11);%Used to calculate at multiple times
jmax = t/dt; %Used to find number of time steps
maxT=zeros(length(t),1);%Used to max temp.
w = 1;
T(length(T)) = 300; %Boundary Condition of Wall Temp = 300C
```

```
for j = 1:jmax(length(t))
    Told = T;
for i=2:N-1
        T(i) = Told(i) + alpha*dt*((Told(i-1)-2*Told(i)+Told(i+1))/(h^2) + q*sin((pi*i*h)/L)/k); %FDE with heat generation
%       T(i) = Told(i) + alpha*dt*((Told(i-1)-2*Told(i)+Told(i+1))/(h^2)); %FDE without heat generation
end
T(1)=T(2); %Boundary Condition used to model insulation at x=0
if j == jmax(w)||jmax(w)==0
maxT(w) = max(T);
w = w+1;
% plot(x,T,'r')
end
end
plot(x,T,'r')
```

A.3 Part B Analytical Method Code

A.3.1 No Decay Heat Generation

```
nmax = 100;% Set maximum number of partial sums
k = 4.5; %Thermal Conductivity [W/mK]
rho = 11000; %Density of UO2 [kg/m^3]
c = 300; %Specific Heat Capcity of UO2 [J/kgK]
alpha = k/(rho*c); %Thermal Diffusivity [m^2/s]
q = (300E6/(2/pi))*0.02; %2% Peaking Power [W]
q1 = (300E6/(2/pi)); %Power at SS condition
Tw = 300; %Tw,b
Tw2 = 420; %Tw,a
% Initialize domain
L = 0.01;
x = linspace(0,L,100);
% Calculate temperature of rod at various times
% t=[5 10 20 40 80];
t = 5;
% t = linspace(0,5,11);
maxT=zeros(length(t),1);%Used to store max temps
for w=1:length(t)
T = 0;
for n=1:nmax
fun = @(x) ((Tw2 + (((L^2)*q1*sin((pi.*x)/(L))/((pi^2)*k))))-Tw).*cos((2*n-1)*pi.*x/(2*L)); %Top integral in eqn 20
top = integral(fun,0,L);

fun2 = @(x) (cos((2*n-1)*pi.*x/(2*L))).^2; %Bottom integral in eqn 20
bottom = integral(fun2,0,L);
Cn = top/bottom;

  T =T +Cn*cos((2*n-1)*pi.*x/(2*L))*exp((-(4*n^2-4*n+1)*(pi^2)*alpha*t(w))/(4*L^2));%Eqn 19
end

T = T + Tw; %Adding initial wall temp back
hold on
plot(x,T,'b--')
maxT(w) = max(T);

end
grid on
% plot(t,maxT,'b--')
% xlabel('Time [s]')
xlabel('x [m]')
ylabel('Temperature [^{o}C]')
```

A.3.2 Decay Heat Generation

```
% clear
% close
% Set maximum number of partial sums
nmax = 100;% Set maximum number of partial sums
k = 4.5; %Thermal Conductivity [W/mK]
rho = 11000; %Density of UO2 [kg/m^3]
c = 300; %Specific Heat Capcity of UO2 [J/kgK]
alpha = k/(rho*c); %Thermal Diffusivity [m^2/s]
q = (300E6/(2/pi))*0.02; %2% Peaking Power [W]
q1 = (300E6/(2/pi)); %Power at SS condition
Tw = 300; %Tw,b
Tw2 = 420; %Tw,a
% Initialize domain
L = 0.01;
x = linspace(0,L,100);
% Calculate temperature of rod at various times
% t=[5 10 20 40 80];
t = 5;
% t = linspace(0,5,10);
maxT=zeros(length(t),1);
 for w=1:length(t)
T = 0;
 for n=1:nmax
fun = @(x) ((Tw2 + (((L^2)*q1*sin((pi.*x)/(L))/((pi^2)*k))))-(Tw+(((L^2)*q*sin(pi*x/L)/((pi^2)*k))+(-L*q.*x/(pi*k))+((L^2)*q/(pi*k))))).*cos((2*n-1)*pi.*x/(2*L));%Top integral eqn 20
top = integral(fun,0,L);
```

```
fun2 = @(x) (cos((2*n-1)*pi.*x/(2*L))).^2;%Bottom integral eqn 20
bottom = integral(fun2,0,L);
Bn = top/bottom;
T =T +Bn*cos((2*n-1)*pi.*x/(2*L))*exp((-(4*n^2-4*n+1)*(pi^2)*alpha*t(w))/(4*L^2));%Transient Solution Eqn 24
end
%Steady State solution terms
term2 = (((L^2)*q*sin(pi*x/L)/((pi^2)*k)));
term3 = (-L*q.*x/(pi*k));
term4 =  ((L^2)*q/(pi*k));
T = T + term2 + term3 + term4 + Tw; %Final Solution
hold on
plot(x,T,'b--')
maxT(w) = max(T);

end
grid on
% plot(t,maxT,'b--')
% xlabel('Time [s]')
xlabel('x [m]')
ylabel('Temperature [^{o}C]')
```