Spring 2025 ENGR 3010 Technical Writing

Final Project

Instructor: Karen Schwelle

Submission Date: May 7, 2025

By: Aidan Moriarty

Contents

1 Introduction

Heat transfer is the process by which energy is moved from one body or substance to another due to a temperature difference. Where this temperature gradient exists, heat will move from regions of high temperature to regions of lower temperature. This process occurs through three mechanisms: conduction, convection, and radiation. Conduction is the transfer of molecular energy through direct contact within or between a solid. Convection, on the other hand, is the transfer of heat between a surface and a fluid, either liquid or gaseous. Radiation is the transfer of energy via electromagnetic waves. Understanding these methods of heat transfer allows us to control and manipulate temperature and energy levels in many practical applications. One such application is cooking. Recipes are the instructions leading to the ideal circumstances for turning ingredients into the desired result. These recipes are usually a result of empirical evidence and the personal experience of the person cooking the food. However, by applying the principles of heat transfer, cooking can be analyzed and optimized with scientific precision. This approach has the potential to move beyond tradition and guesswork, offering a systematic path to improved consistency, efficiency, and quality in food preparation. Research into this kind of approach has been lacking thus far, although chef Peter Hertzmann notes that ”How heat travels from its source into our food is important. The physics of cooking teaches us to choose our heat transfer agents and cooking methods wisely. If we do so, we can successfully place food on our table that is cooked exactly as we desire [1].” The central questions addressed in this experiment are: How can the transient temperature distribution within a hot dog during grilling be modeled, and what setup yields optimal cooking efficiency while ensuring the center reaches a safe and desirable temperature?

The general approach to this question is to model the radiation from the charcoal on the grill, convection from the surrounding air, and conduction within the hot dog to reflect all stages of the heat transfer process occurring in the system. Ideal conditions are determined by optimizing the hot dog’s position relative to the coals, the size of the grill, and the cooking time. Initial conditions and properties are defined as follows:

- The initial temperature of the hot dog is 10 °C throughout.

- The surface temperature must not exceed 100 °C at any time.
- The final temperature of the centerline of the hot dog should be 68 °C.
- Assume the air surrounding the hot dog to have a temperature of 250°C.
- Assume a coal temperature of 450°C.
- Assume an emissivity of 0.8 for the coals and 0.45 for the hot dog.
- $\rho_{\text{hotdog}} = 880 \frac{\text{kg}}{\text{m}^3}$
- $k_{\text{hotdog}} = 0.52 \frac{\text{W}}{\text{m} \cdot \text{K}}$
- $c_{\text{hotdog}} = 3350 \frac{\text{J}}{\text{kg} \cdot \text{K}}$
- $D_{\text{hotdog}} = 2.54\text{cm}$

The general process for identifying the ideal barbecue conditions was to first determine the total heat transfer coefficient, h. This was done by applying a root finding algorithm in MATLAB and iterating over all possible Biot numbers until a Biot number resulting in a centerline of 68 °C and a surface of 100 °C is identified. Once the h value is calculated from this Biot number, both an analytical and numerical method are applied in order to simulate the temperature distribution in the hot dog over time and determine the final cooking time. The convective heat transfer coefficient is found by using forced convection correlations, such that the radiative heat transfer coefficient can be calculated by finding the difference between the total and convective heat transfer coefficients. The radiative heat transfer coefficient is required to determine the optimal placement and height of the hot dog relative to the grill, which is used by determining the view factor between both surfaces.

2 Calculation of Heat Transfer Coefficients

2.1 Optimal Total Heat Transfer Coefficient. It was necessary to first determine the overall heat transfer coefficient based on the desired final temperatures of the hot dog. A MATLAB program was used to iterate over a set of Biot numbers until the corresponding values of ζ_1 and C_n that resulted in the centerline reaching 68 °C and the surface reaching 100 °C. The relationship between the Biot number, ζ_1, and C_n are shown below in Equations 5.50a, 5.50b, and 5.50c [2]. Once the optimal Biot number was determined, Equation 1 was used in order to calculate the optimal total heat transfer coefficient, which was found to be 16.665 $\frac{\mathrm{W}}{\mathrm{m^2 \cdot K}}$.

$$\mathrm{Bi} = \frac{h_{\mathrm{opt}} \cdot r_{\mathrm{hotdog}}}{k_{\mathrm{hotdog}}} \tag{1}$$

For an infinite cylinder in dimensionless form, the temperature is

$$\theta^* = \sum_{n=1}^{\infty} C_n \exp(-\zeta_n^2 Fo) J_0(\zeta_n r^*) \tag{5.50a}$$

where

$$\mathrm{Fo} = \frac{\alpha \cdot t}{r_0^2}$$

and

$$C_n = \frac{2}{\zeta_n} \cdot \frac{J_1(\zeta_n)}{J_0^2(\zeta_n) + J_1^2(\zeta_n)} \tag{5.50b}$$

and the discrete values of ζ_n are positive roots of the transcendental equation

$$\zeta_n \frac{J_1(\zeta_n)}{J_0(\zeta_n)} = Bi \tag{5.50c}$$

2.2 Convective Heat Transfer Coefficient. Forced convection relationships between the hot air rising from the coals were used to calculate the convective heat transfer coefficient. This process started by determining the velocity of the hot air at the coals, shown below in Equation 2. Then, the continuity equation, shown below as Equation 3, was used to calculate the velocity of the hot air surrounding the hot dog. Upon calculating this velocity, it was then possible to determine the Reynolds number of the fluid (Equation 4), then the Nusselt number (Equation 5), and finally the convective heat transfer coefficient (Equation 6). This value was found to be 11.736 $\frac{\mathrm{W}}{\mathrm{m^2 \cdot K}}$.
The characteristic velocity equation from the free convection of the coals is given as:

$$u_{\mathrm{coals}}^2 = g\beta(T_s - T_\infty)D \tag{2}$$

where the temperature of the coals T_s = 450 °C, the ambient outside temperature T_∞ = 20 °C, and D is the diameter of a single coal.

The continuity equation between the momentum of the air at the coals and at the hot dog is given as:

$$\rho_{\text{coals,air}} A_{\text{flow}} u_{\text{coals}} = \rho_{\text{hotdog,air}} A_{\text{flow,grill}} u_{\text{hotdog}} \tag{3}$$

where ρ represents the density of air at each location, and A_{flow} is $\frac{1}{2}$ of A_{grill}.

The Reynolds number is then given by:

$$\text{Re}_{\text{D}} = \frac{u_{\infty} D}{\nu} \tag{4}$$

Where ν is the ratio of convective heat transfer to conductive heat transfer at the hot dog in the surrounding air.

The average Nusselt number for forced convection over a cylinder is given by:

$$\overline{\text{Nu}}_{\text{D,hd}} = 0.3 + \left[\frac{0.62\,\text{Re}_{\text{D,hotdog}}{}^{1/2}\,\text{Pr}_{\text{film}}{}^{1/3}}{\left(1 + \left(\frac{0.4}{\text{Pr}_{\text{film}}}\right)^{2/3}\right)^{1/4}} \right] \left(1 + \left(\frac{\text{Re}_{\text{D,hotdog}}}{282000}\right)^{5/8}\right)^{4/5} \tag{5}$$

From which the average convective heat transfer $\overline{h}_{\text{conv}}$ can be calculated:

$$\overline{h}_{\text{conv}} = \frac{\overline{\text{Nu}}_{\text{D,hd}} \cdot k_{\text{air}}}{D} \tag{6}$$

Where k_{air} is the thermal conductivity of the air at the film temperature between the hot dog and the surrounding air.

2.3 Radiation Heat Transfer Coefficient. The radiation heat transfer coefficient is determined by subtracting the average convective heat transfer coefficient from the calculated optimal total heat transfer coefficient. As shown in Equation 7, the value of the radiation heat transfer coefficient is calculated to be 4.929 $\frac{\text{W}}{\text{m}^2 \cdot \text{K}}$.

$$16.665 - 11.736 = 4.929 \frac{\text{W}}{\text{m}^2 \cdot \text{K}} \tag{7}$$

3 Calculating Optimal Hot dog Placement

Once the radiative heat transfer coefficient is calculated, it is now possible to determine the optimal location of the hot dog on the grill. Shown below in Fig. 1 is a diagram of the system being examined. Given that radiation is being emitted from a rectangle parallel with a cylinder, shown in

Fig. 2, Equation 8 is utilized to calculate the view factor between the hot dog and the surface of the grill [2].

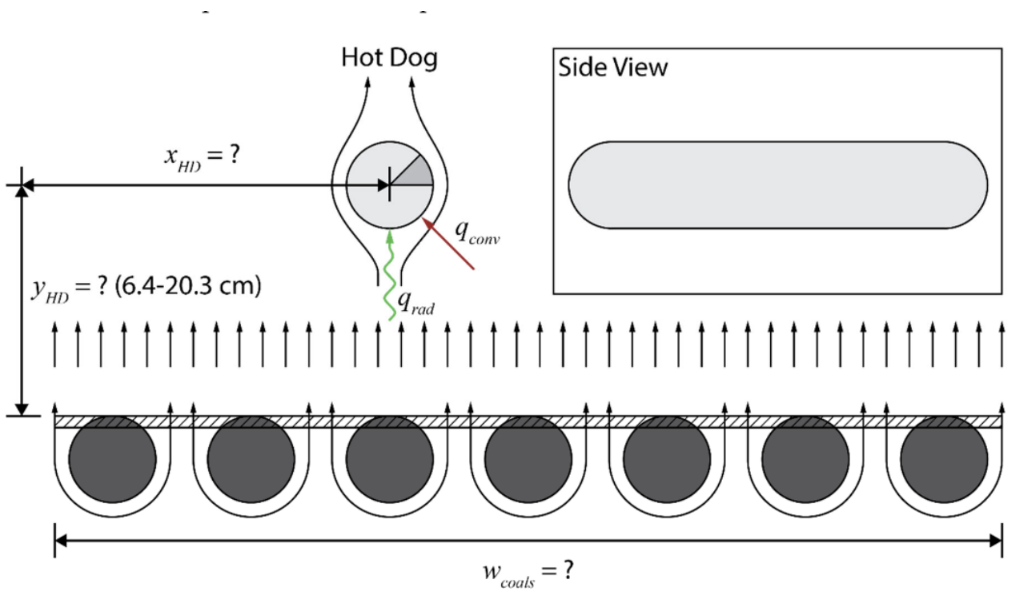

Figure 1 System schematic

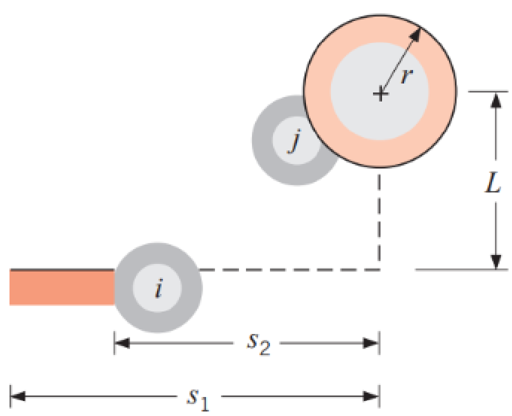

Figure 2 Diagram of view factor between a cylinder and parallel rectangle [2]

The view factor for a rectangle parallel with a cylinder is shown below:

$$F_{12} = \frac{r}{s_1 - s_2}\left[\tan^{-1}\left(\frac{s_1}{L}\right) - \tan^{-1}\left(\frac{s_2}{L}\right)\right] \tag{8}$$

Using the assumptions that $T_{\text{amb}} << T_{\text{coals}}$ and $T_s << T_{\text{coals}}$, it is possible to say that $\rho G << \varepsilon E_b$, which means that $J_{\text{coals}} = \varepsilon E_b$. By applying the definition of radiosity, a formula is derived for the driving potential per space resistance, shown below in Equation 9. A write up of this derivation is included in Appendix A. Below, the number 2 is used to signify the surface of the hot dog, and 1 is used to signify the coals.

$$q_{1-2} = A_1 F_{12} \varepsilon_1 \sigma T_1^4 \tag{9}$$

Now the amount absorbed at the hot dog can be written as,

$$q_{\text{absorbed}} = \alpha_2 \epsilon_2 A_1 F_{12} \sigma T_1^4$$

For a diffuse, gray hot dog: $\varepsilon_2 = \alpha_2$, and using reciprocity between surfaces:

$$q_{\text{absorbed}} = \varepsilon_1 \varepsilon_2 A_2 F_{12} \sigma T_1^4$$

Now emission from hotdog over the entire surfacer is given by:

$$q_{\text{emitted}} = \varepsilon_2 A_2 \sigma T_2^4$$

Multiplying and dividing by $\varepsilon_1 F_{21}$:

$$q_{\text{emitted}} = \varepsilon_1 \varepsilon_2 A_2 F_{21} \sigma [\frac{T_2}{(\varepsilon_1 F_{21})^{1/4}}]^4$$

Where

$$T_2^* = \frac{T_2}{(\varepsilon_1 F_{21})^{1/4}} \tag{10}$$

So that

$$q_{\text{radiation}} = \text{q}_\text{a}\text{bsorbed} - q_{\text{emitted}} = \varepsilon_1 \varepsilon_2 A_2 F_{21} \sigma [T_1^4 - T_2^{*^4}]$$

Linearizing,

$$q_{\text{radiation}} = \varepsilon_1 \varepsilon_2 A_2 F_{21} \sigma [T_1 + T_2^*][T_1^2 + T_2^{*^2}][T_1 - T_2^*] \tag{11}$$

finally,

$$h_{\text{rad}} = \varepsilon_1 \varepsilon_2 F_{21} \sigma [T_1 + T_2^*][T_1^2 + T_2^{*^2}] \tag{12}$$

Using this relationship and Equation 8, a MATLAB program was created that iterates over the length over the grill and calculates the height required to achieve the desired h_{rad}. A plot of the optimal locations along the length of a 0.4 m grill is shown in Fig. 3.

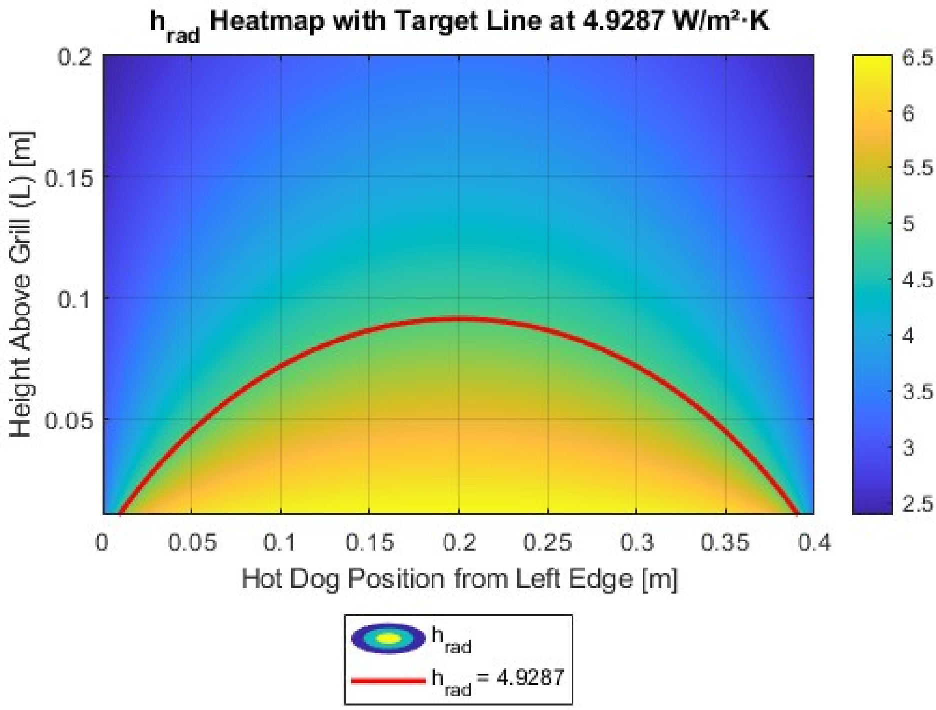

Figure 3 Optimal height of hot dog above grill to achieve target h_{rad}

Here, it is seen that the optimal height above the grill follows an arc above it, with the farthest edges have the lowest value of L, while the center of the grill has a maximum L of around 0.9 m.

4 Analytical Solution

In order to calculate the analytical solution, an approximate one-term solution to Equation 5.50a was determined using the calculated values of C_1 and ζ_1, and the temperature distribution shown in Equation 13, to calculate the final time taken to cook the hot dog. This time was determined to be 455.6 s.

$$\theta_0^* = \frac{T_s - T_\infty}{T_i - T_\infty} \tag{13}$$

Figure 4 plots the temperature at the centerline and surface of the hot dog over time. In Fig. 5, a temperature distribution in the hot dog at different times is plotted over the radial distance from the centerline to the surface.

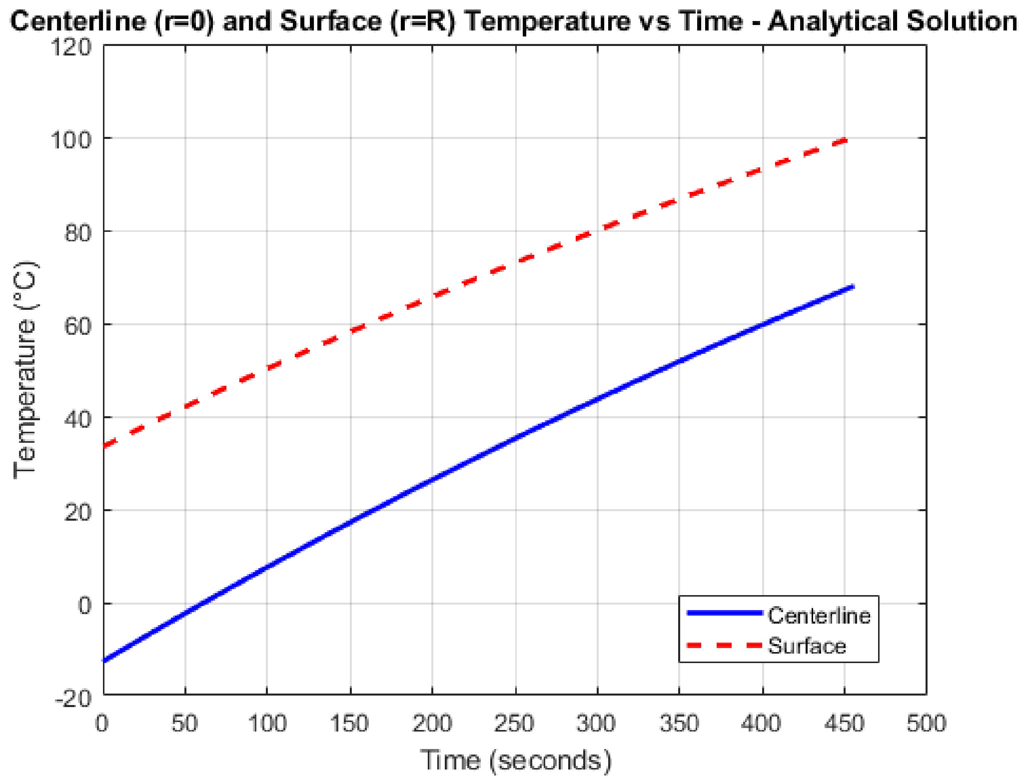

Figure 4 Temperature of the centerline and the surface versus time

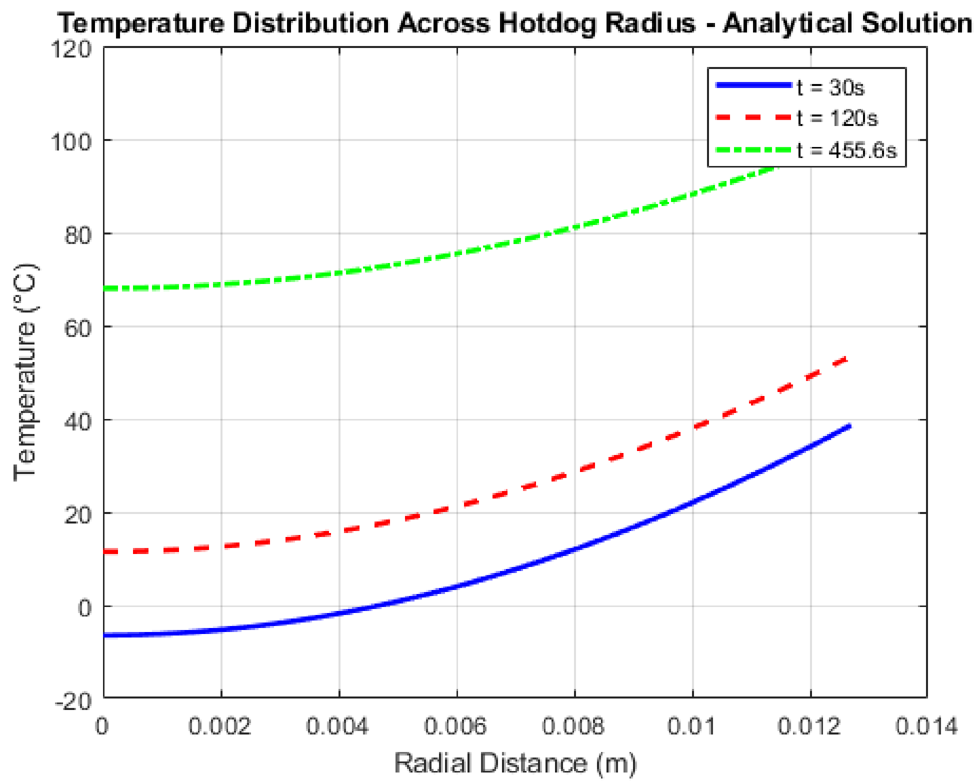

Figure 5 Temperature versus radial distance at three times

When analyzing the results, at the final time the Fourier number is sufficiently large, so the one term approximation is more accurate. At shorter time steps however the plots are shown to be very inaccurate, as seen by the negative temperatures at early times in both plots. This is due to the fact that for small times, higher-order Bessel function terms are non-negligible, invalidating the one-term solution at those times. In Fig. 4, the initial growth of the lines is almost linear, but in reality the temperature at the centerline should be increasing slowly intially a conduction slowly spreads from the surface of the hot dog to the center. In Fig. 5, the final temperature curve is sufficiently accurate at the final time of 455.6s, shown in green, but again at a shorter time of 30s the model predicts a centerline temperature of -10 °C and a surface temperature of around 40 °C, which is impossible given the initial conditions provided. So as a model, the analytical solution is a good predictor for what the final result will look like at a longer time, but intially lacks in accuracy.

5 Numerical Solution

The numerical solution applies an explicit finite differencing approach. The hot dog is modeled as a series of nodes beginning at the centerline and proceeding outwards to the surface, modeling the heat transfer at each location in the hot dog. There are three different types of nodes defined in this model: the surface node, which interacts with one nodal neighbor and the convection from the air surrounding the hot dog; the inner nodes, which each interact with two other nodal neighbors; and the center node, which only interacts with one nodal neighbor due to the symmetry in the cylinder. Beginning with the 1D, transient heat diffusion equation in finite differencing form,

$$\frac{1}{\alpha}\frac{T_m^{i+1} - T_m^i}{\Delta t} = \frac{T_{m+1}^i + T_{m-1}^i - 2T_m^i}{(\Delta x)^2} \tag{14}$$

Rearranging:

$$T_m^{i+1} = \mathrm{Fo}(T_{m-1}^i + T_{m+1}^i) + (1 - 2\mathrm{Fo})T_m^i$$

for $i = 0, 1, 2, \ldots$ and $m = 1, 2, 3, \ldots$ and

$$\mathrm{Fo} = \frac{\alpha \Delta t}{(\Delta x)^2}$$

Which represents the finite difference fourier number.

In order to determine the temperature of each node, beginning with an energy balance around each nodal region:

$$\sum q(i) = \dot{e}\Delta V \tag{15}$$

Where $q(i)$ is the heat transfer in node i, $\dot{e}$ is the change in energy with time per unit volume, and ΔV is the change in volume.

5.1 Surface Node (m = M). Applying energy balance:

$$q_{m-1/2} + q_{\text{conv}} = \dot{e}\,\Delta V$$

$$k2\pi(M-1/2)\Delta r H\frac{T_{M-1}-T_M}{\Delta r} + k2\pi M\Delta r H h(T_\infty - T_M) = \rho c(\frac{T_M^{i+1}-T_M^i}{\Delta t})(2\pi M\Delta r^2 H)$$

$$\frac{k\Delta t}{\rho c m\Delta r^2}(MT_{M-1} - MT_M - \frac{1}{2}T_{M-1} + \frac{1}{2}T_M) + \frac{mh\Delta t}{\rho c m\Delta r^2}(T_\infty - T_M) + T_M^i = T_M^{i+1}$$

$$\frac{\text{Fo}}{M}[(M-\frac{1}{2})T_{M-1} + (\frac{1}{2}-M)T_M] + \frac{h\Delta t}{\rho c\Delta r}(T_\infty - T_M) + T_M^i = T_M^{i+1}$$

$$\text{Fo}(1-\frac{1}{2M})T_{M-1} + \text{Fo}(\frac{1}{2M}-1)T_M + \frac{h\Delta t}{\rho c\Delta r}(T_\infty - T_M) + T_M^i = T_M^{i+1} \tag{16}$$

Equation 16 represents the temperature at surface node M at the next time step, based on the calculation of the temperature of the surface node at the current timestep added to the effects on the future node.

5.2 Inner Node (m). Applying energy balance:

$$q_{m-1/2} + q_{m+1/2} = \dot{e}\,\Delta V$$

$$k2\pi(m-\frac{1}{2})\Delta r H(\frac{T_{m-1}-T_m}{\Delta r}) + k2\pi(m+\frac{1}{2})\Delta r H(\frac{T_{m+1}-T_m}{\Delta r}) = \rho c(\frac{T_m^{i+1}-T_m^i}{\Delta t})(2\pi m\Delta r^2 H)$$

$$\frac{k\Delta t}{\rho c\Delta r^2 m}[(m-\frac{1}{2})(T_{m-1}-T_m) + (m+\frac{1}{2})(T_{m+1}-T_m)] + T_m^i = T_m^{i+1}$$

$$\text{Fo}[(1-\frac{1}{2m})T_{m-1} - 2T_m + (1+\frac{1}{2m})T_{m+1}] + T_m^i = T_m^{i+1} \tag{17}$$

Equation 17 represents the temperature at node m at the next time step, based on the calculation of the temperature of the inner node at the current timestamp added to the effects on the future node.

5.3 Center Node (m = 0). Applying energy balance:

$$q_{1/2} = \dot{e}\,\Delta V$$

$$k2\pi(\frac{\Delta r}{2})H(\frac{T_1 - T_0}{\Delta r}) = \rho c(\frac{T_m^{i+1} - T_m}{\Delta t})(\frac{\pi \Delta r^2}{2})H$$

$$2\mathrm{Fo}T_1 + (1 - 2\mathrm{Fo})T_0 = T_0^{i+1} \tag{18}$$

Equation 18 represents the temperature at the center node at the next time step, based on the calculation of the temperature of the center node at the current timestamp added to the effects on the future node.

5.4 Stability Criteria. A stability criterion for the time steps used in the numerical solution was also calculated as required for the explicit method. This is shown in Equation 19 below.

$$\Delta t \leq \frac{1}{2\alpha}(\Delta r)^2 \tag{19}$$

where

$$\Delta r = \frac{r}{m}$$

The plots of the results for the numerical method are shown below. Figure 6 plots the temperatures at the centerline and the surface of the hot dog over time. Figure 7 compares the temperature in the hot dog along its radial position, measured at three different times.

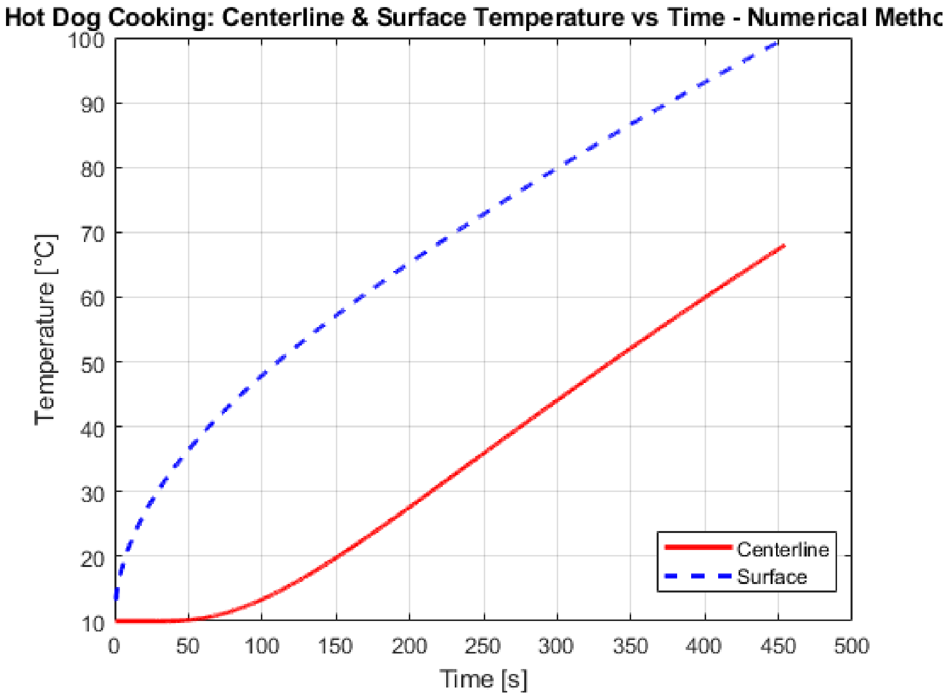

Figure 6 Temperature of the centerline and the surface verus time

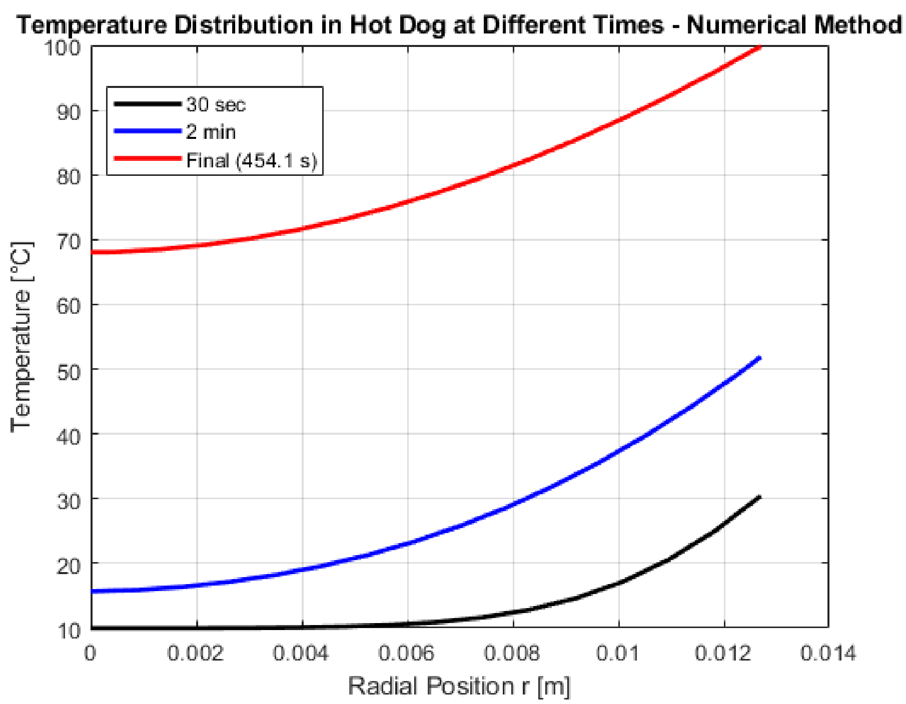

Figure 7 Temperature versus radial distance at three times

The numerical solution appears much more accurate compared to the analytical plots. In Fig 7, we see the temperature beginning at the correct value of 10 °C, and then gradually increasing overtime, consistent with the result expected. At the final time of 454.1 s, the centerline of the hot dog is at 68 °C and the surface is at 100 °C, fulfilling our conditions. In Fig 6, again the initial start is at 10 °C for both the surface and the centerline, we see the expected trend of the surface of the hot dog rapidly heating as it is exposed to the convection and radiation, while the centerline of the hot dog takes longer to heat up as conduction is spread through the nodes. Overall, the quality of this model is very high as a predictor and provides an accurate representation of exactly what to expect when cooking the hot dog due to no longer having to rely on a one term approximation and directly calculating the relationship between points in the hot dog.

6 Discussion of Results

Both solutions met the final temperature constraints, however the numerical solution was overall much more consistent and accurate than the analytical solution. The analytical one-term approximate solution shown in Figs 4 and 5 successfully approximate the heat distribution in the hot dog, but accuracy drops off quite a bit at low Fourier numbers. They are both successful at modeling close to the final time, with almost the same final trends as the numerical method. The graphs for the analytical method have a very similar shape to those in the numerical method, but begin at temperatures below zero, which is incorrect. The numerical method shown in Figs 6 and 7 was overall extremely precise with its simulation. The trends are exactly what are expected from the problem, and the ability to customize variables such as amount of nodes in the hot dog allows for even more exact calculations.

In order to validate the model as an accurate predictor of optimal conditions and cooking time, one hot dog was barbecued experimentally. The first attempt at this process was to recreate the constraints identified prior to creating the model. However, this was found to be much more difficult than initially realized. First, it was difficult to achieve a uniform temperature of 450 °C along the coals, because they were constantly heating up once the grill was lit. This issue led to the decision that rather than stop the experiment and reset the grill, the research would continue under the new coal condition of 550 °C. Another issue was that while two different temperature sensors were brought to measure the temperature of the hot dog, a regular thermometer capable of measuring the ambient air on the day of the experiment was not brought, meaning that it was necessary to rely on the somewhat inaccurate temperature provided by the national weather service of the day in question. Also, the initial temperature of the hot dog was around 15-20 °C, not the given initial value of 10 °C. This was due to the fact that the hot dog package was not placed in an insulated temperature environment, but next to the grill exposed to the ambient temperature.

Other than these limitations, the experiment was able to move forward without too many more issues. One wire thermometer was inserted into the hot dog to measure the centerline temperature, and one laser thermometer was used to measure the surface temperature. The resulting time needed to cook the hot dog under the conditions given was around 410 seconds, which was determined to be an acceptable result once the model was redone to better suit the temperatures of the day. One interesting result of the experiment was our surprise at the resulting hot dog that was deemed ”perfectly barbecued” by our initial constraints, the centerline was cooked very nicely but most who tried to experimental hot dogs determined that the surface could be a little crispier. Overall, the model did an extremely good job at predicting the experimental growth of the centerline temperature through continued exposure to conduction and radiation, simulating the exponential growth that occurs as each node along the hot dog continues to heat up. It was a very satisfying experiment and resulted in many pleasant memories.

There are more opportunities for further research related to the modeling of heat transfer in food now that the accuracy of the model has been validated. While this study focused entirely on a cylindrical geometry, I am interested in how other types of geometries would be computed. Many of the assumptions made throughout the study were unique to a cylinder, so I am interested in what other assumptions exist for shapes such as spheres or rectangular shapes. One limitation in that avenue of research is the relatively perfect geometry of a hot dog, being that it is perfectly cylindrical by design, whereas if I were to apply the model to another item of meat such as a chicken breast or a steak, the geometry is not easily identified. My initial assumption would then have to be that the geometry is to be assumed as something else, thereby potentially reducing the accuracy of the model. Similarly, more complex geometries such foods with multiple components could be modeled using multi-domain methods or internal boundary layers to represent different materials in thermal contact. This opens up opportunities for modeling heterogeneous cooking environments, such as baking, where moisture loss, crust formation, and surface browning all interacting dynamically. I am confident that the same overall principles and processes can be applied to any object, not just food, undergoing a comprehensive heating involving all three types of heat transfer. The MATLAB code included in Appendix B provides a sufficient blueprint that can be modified and customized for many more objects of study using the same process. I am also interested in comparing the results of the model with recipes that require a certain level of cooking to food. I believe that the model could be improved with a more user friendly user interface and then provided to those wishing to achieve a certain level of cook, allowing them to customize their own cooking setup, and then be provided with the optimal conditions for cooking.

7 Conclusion

This study successfully applied heat transfer principles to model and analyze the cooking of a hot dog using both analytical and numerical methods. A model was developed and customized to reflect the real world conditions of barbecuing a hot dog and was able to predict the time to cook and necessary height above the grill required to achieve the ideal cooking conditions. The model created is versatile and is applicable to a wide range of geometries and cooking methods, and serves as a crucial first step towards further research into building more comprehensive food cooking models. There are many future opportunities in simulating ideal cooking conditions that can lead to improved safety and precision in food preparation. While there were many assumptions that limited the scope of this model, future studies could improve accuracy in a variety of ways. Overall, this project demonstrated that even a simple food item can serve as a powerful vehicle for understanding complex thermal processes, and that engineering tools can meaningfully inform everyday practices like cooking.

References

[1] Hertzmann, P., 2006, "Cooking 101: Heat Transfer," https://www.hertzmann.com/articles/2006/heat/

[2] Bergman, T., Lavine, A., Incropera, F., and DeWitt, D., 2011, *Fundamentals of heat and mass transfer (7th ed.)*, John Wiley & Sons.

A Derivation of Radiation Exchange

Let Radiosity, J, be defined as the following:

$$J = E + \rho G \tag{20}$$

where E is the emissive power emitted by the surface, ρ is the reflectivity of the surface, and G is the irradiation of the surface. Let $q^{"}$ be the net radiant flux leaving the surface, equal to the heat flux supplied to the surface, such that just above the surface:

$$q^{"} = J - G \tag{21}$$

and just below the surface:

$$q^{"} = E - \alpha G \tag{22}$$

from the fact that $J = E + \rho G$ and $\rho = 1 - \alpha$.

From Equation 20:

$$G = \frac{J - E}{\rho} = \frac{J - \varepsilon E_b}{1 - \varepsilon}$$

Plugging this into Equation 21:

$$q^{"} = J - \frac{J - \varepsilon E_b}{1 - \varepsilon} = \frac{\varepsilon(E_b - J)}{1 - \varepsilon}$$

Or, for a given surface 'i':

$$q_i = q_i^{"} A_i = \frac{E_{b,i} - J_i}{\frac{1-\varepsilon_i}{\varepsilon_i A_i}} \tag{23}$$

For an arbitrary enclosure,

$$A_i G_i = \sum_{j=1}^{N} A_j F_{ji} J_i$$

from reciprocity,

$$A_j F_{ji} = A_i F_{ij}$$

so

$$G_j = \sum_{j=1}^{N} F_{ij} J_i$$

subbing into Equation 21,

$$q_i^{"} = J_i - \sum_{j=1}^{N} F_{ij} J_i$$

Because

$$\sum_{j=1}^{N} F_{ij} = 1$$

we can write

$$q_i^{"} = \sum_{j=1}^{N} F_{ij} J_i - \sum_{j=1}^{N} F_{ij} J_j$$

or, multiplying by A_i,

$$q_i = \sum_{j=1}^{N} A_i F_{ij} (J_i - J_j)$$

which can be rewritten as

$$q_i = \sum_{j=1}^{N} \frac{J_i - J_j}{(A_i F_{ij})^{-1}}$$

Equating with Equation 23,

$$\frac{E_{b,i} - J_i}{[\frac{1-\varepsilon_1}{\varepsilon_i A_i}]} = \sum_{j=1}^{N} \frac{J_i - J_j}{(A_i F_{ij})^{-1}} \tag{13.21}$$

B MATLAB code

```
% Aidan Moriarty - Heat Transfer Project
clear;


%% GIVEN PROPERTIES
T_coals = 450+273.15;
T_inf = 250;
T_2 = 55; % avg surface temp
T_0 = 10; % initial hd temp
D_hd = 0.0254; % [m]
r = D_hd/2;
e1 = 0.8;
e2 = 0.45;
sigma = 5.67e-8;
k_hd = 0.52;
rho_hd = 880;
cp_hd = 3350;
alpha = k_hd / (rho_hd * cp_hd);


%% PROPERTIES FOR COEFFICIENT CALCULATIONS

T_amb = 20; % [C]
T_film_coals = (T_coals + T_amb)/2; % [C]
a_grill = 0.4; %[m]



%% CALCULATE OPTIMAL TOTAL COEFFICIENT
% --- PARAMETERS ---
Tc_target = 68;
Ts_target = 100;

theta_c_target = (Tc_target - T_inf) / (T_0 - T_inf);
theta_s_target = (Ts_target - T_inf) / (T_0 - T_inf);
```

```
tol = 0.00001;

% --- Bi Guessing ---
Bi = 0.3;
max_bi_iter = 500;
dt = 0.1;                % seconds
max_time_iter = 10000;

zeta_vals = linspace(0.01, 10, 10000);
success = false;

best_err = inf;
best_result = struct();

for bi_iter = 1:max_bi_iter
    trans_func = @(zeta) zeta .* (besselj(1,zeta) ./ besselj(0,zeta)) - Bi;
    f_vals = trans_func(zeta_vals);
    sign_changes = find(diff(sign(f_vals)));

    if isempty(sign_changes)
        Bi = Bi + 0.001;
        continue
    end

    idx = sign_changes(1);
    zeta1 = fzero(trans_func, [zeta_vals(idx), zeta_vals(idx+1)]);
    J0 = besselj(0, zeta1);
    J1 = besselj(1, zeta1);
    C1 = 2 * J1 / (zeta1 * J0^2 + J1^2);

    Fo = 0;
    t = 0;
    for time_iter = 1:max_time_iter
        Fo = Fo + alpha * dt / r^2;
        t = t + dt;
        theta_c = C1 * exp(-zeta1^2 * Fo);
        theta_s = C1 * besselj(0, zeta1) * exp(-zeta1^2 * Fo);
        Tc = T_inf + theta_c * (T_0 - T_inf);
```

```
        Ts = T_inf + theta_s * (T_0 - T_inf);
        err_c = abs(theta_c - theta_c_target);
        err_s = abs(theta_s - theta_s_target);
        total_err = err_c + err_s;

        if total_err < best_err
            best_err = total_err;
            best_result = struct('Bi', Bi, 'zeta1', zeta1, 'C1', C1, ...
                                 'Tc', Tc, 'Ts', Ts, 't', t, 'Fo', Fo);
        end

        % New: only accept convergence if both are individually below tol
        if err_c < tol && err_s < tol
            break
        end
    end
    Bi = Bi + 0.001;
end

h_opt = best_result.Bi * k_hd / r;
disp(h_opt);
Bi = best_result.Bi;
% % --- Final Output ---
% fprintf("\ n   BEST MATCH FOUND:\n");
% fprintf("Bi = %.4f |        = %.4f | h_opt = %.2f W/ m   K\n", ...
%     best_result.Bi, best_result.zeta1, best_result.Bi * k_hd / r);
% fprintf("Tc = %.2f C | Ts = %.2f C | Time = %.1fs | Total Error = %.4f\n", ...
%     best_result.Tc, best_result.Ts, best_result.t, best_err);
% disp(Fo);

% Function to solve for zeta_1
function zeta1 = find_zeta1(Bi)
    % Define the transcendental equation: zeta * J1(zeta)/J0(zeta) - Bi = 0
    f = @(z) z * besselj(1, z) / besselj(0, z) - Bi;
    % Initial guess for zeta_1 (first root is typically small)
    zeta_guess = 0.1;
    zeta1 = fzero(f, zeta_guess);
end
```

```
% Function to compute C_1
function C1 = compute_C1(zeta1)
    J0_zeta = besselj(0, zeta1);
    J1_zeta = besselj(1, zeta1);
    C1 = 2 / zeta1 * J1_zeta / (J0_zeta^2 + J1_zeta^2);
end

 % Solve for zeta_1
    zeta1 = find_zeta1(Bi);

    % Compute C_1
    C1 = compute_C1(zeta1);

% Target theta_star at centerline
    theta_star_center = (68 - 250) / (10 - 250);

    % Compute Fo for centerline temperature
    Fo = log(theta_star_center / C1) / (-zeta1^2);

    % Final time
    t_final = Fo * r^2 / alpha;



%% CALCULATE CONVECTION COEFFICIENT

rho_coals_air = 0.6964; %[kg/m^3]
rho_hotdog_air = (0.8711+0.7740)/2; %[kg/m^3]
a_coals = 0.5*a_grill;
d_coals = 0.05; %[m]
beta = 1/(T_film_coals+273.15);
g = 9.81; %[m/s^2]
v_film = ((26.41+32.39)/2)*10^(-6);
k_film = ((33.8+37.3)/2)*10^(-3);
Pr_film = (0.690+0.686)/2;
disp(beta);
velocity_coals = sqrt(g*beta*(T_coals-T_amb)*d_coals);
```

```
velocity_hd = (rho_coals_air*a_coals*velocity_coals)/(rho_hotdog_air*a_grill);
Re_D_hd = (velocity_hd*r*2)/v_film;
Nu_D_avg_hd = 0.3 + ...
    ((0.62*(Re_D_hd^(1/2))*(Pr_film^(1/3)))*(1+((0.4/Pr_film)^(2/3)))^(-0.25)*(1+((Re_D_hd/28
    %eq 7.54

h_conv_avg = (Nu_D_avg_hd * k_film)/(2*r);

disp(h_conv_avg);

%% CALCULATE RADIATION COEFFICIENT

T1 = T_coals;  % [K]
T2 = T_2+273.15;      % [K]
h_rad_target = h_opt-h_conv_avg;  % [W/ m   K ]
tolerance = 0.01;

% --- GRID SETUP ---
x_range = linspace(0, a_grill, 300);       % hot dog positions
L_range = linspace(0.01, 0.2, 300);        % heights
[X, L] = meshgrid(x_range, L_range);
h_rad_map = zeros(size(X));

% --- CALCULATE h_rad OVER GRID ---
for i = 1:length(L_range)
    for j = 1:length(x_range)
        x_pos = x_range(j);
        L_val = L_range(i);

        s1 = x_pos;
        s2 = a_grill - x_pos;

        F_12 = (r / (s1 + s2)) * (atan(s1 / L_val) + atan(s2 / L_val));
        F_21 = (a_grill / (2 * pi * r)) * F_12;

        h_rad_map(i, j) = e1 * e2 * F_21 * sigma * (T1 + T2) * (T1^2 + T2^2);
    end
end
```

```
% --- PLOT HEATMAP WITH h_rad TARGET LINE ---
figure;
contourf(X, L, h_rad_map, 100, 'LineColor', 'none');
hold on;

% Red contour line where h_rad = target    tolerance
contour(X, L, h_rad_map, [h_rad_target h_rad_target], 'r', 'LineWidth', 2);

xlabel('Hot Dog Position from Left Edge [m]');
ylabel('Height Above Grill (L) [m]');
title(['h_{rad} Heatmap with Target Line at ', num2str(h_rad_target), ' ...
    W/ m   K ']);
colorbar;
grid on;
legend('h_{rad}', ['h_{rad} = ', num2str(h_rad_target)], 'Location', ...
    'southoutside');

%% ANALYTICAL SOLUTION

t = linspace(0, t_final, 100);
% Plotting centerline and surface temperature vs time
T_center_t = (T_0 - T_inf) * C1 * exp(-zeta1^2 * alpha * t / r^2) + T_inf;
T_surface_t = (T_0 - T_inf) * C1 * exp(-zeta1^2 * alpha * t / r^2) * ...
    besselj(0, zeta1 * 1) + T_inf;

figure(5);
plot(t, T_center_t, 'b-', 'LineWidth', 2, 'DisplayName', 'Centerline');
hold on;
plot(t, T_surface_t, 'r--', 'LineWidth', 2, 'DisplayName', 'Surface');
xlabel('Time (seconds)');
ylabel('Temperature ( C )');
title('Centerline (r=0) and Surface (r=R) Temperature vs Time - Analytical ...
    Solution');
legend;
hold off;

% Plot spatial temperature distribution
```

```
x = linspace(0, r, 100);
r_star = x / r;
t_1 = 30;
t_2 = 120;
T_r_t1 = (T_0 - T_inf) * C1 * exp(-zeta1^2 * alpha * t_1 / r^2) * besselj(0, ...
    zeta1 * r_star) + T_inf;
T_r_t2 = (T_0 - T_inf) * C1 * exp(-zeta1^2 * alpha * t_2 / r^2) * besselj(0, ...
    zeta1 * r_star) + T_inf;
T_r_tf = (T_0 - T_inf) * C1 * exp(-zeta1^2 * alpha * t_final / r^2) * ...
    besselj(0, zeta1 * r_star) + T_inf;

figure(6);
plot(x, T_r_t1, 'b-','LineWidth', 2, 'DisplayName', 't = 30s');
hold on;
plot(x, T_r_t2, 'r--','LineWidth', 2,'DisplayName', 't = 120s');
plot(x, T_r_tf,'g-.','LineWidth', 2, 'DisplayName', sprintf('t = %.1fs', ...
    t_final));
xlabel('Radial Distance (m)');
ylabel('Temperature ( C )');
title('Temperature Distribution Across Hotdog Radius - Analytical Solution');
legend;
hold off;




%% NUMERICAL SOLUTION

% Parameters
N = 30;               % number of nodes
dr = (D_hd/2)/N;
dt = (1/2) * (dr^2 / alpha);  % Stability condition
Fo = alpha * dt / dr^2;

% Initialization
T = ones(N,1)*10;      % start at 10 C
T_new = T;
time = 0;
```

```
times = [];
center_T = [];
surface_T = [];
r = linspace(0, r, N);    % radial positions

% For spatial snapshots
snapshot_30s = [];
snapshot_2min = [];
snapshot_final = [];

while T(1) < 68
    for n = 1:N
        if n == 1
            % Centerline symmetry BC
            T_new(n) = 2*Fo*T(n+1) + (1 - 2*Fo)*T(n);
        elseif n == N
            % Surface node with convection + radiation
            T_new(n) = Fo*(2-(1/n))*T(n-1) + ...
                            (Fo*((1/n)-2)+1)*T(n) + ...
                            ((2*h_opt*dt)/(rho_hd*cp_hd*dr))*(T_inf-T(n));
        else
            % Internal nodes
            T_new(n) = Fo*(1 + (1/(2*n)))*T(n+1) + ...
                       (1 - 2*Fo)*T(n) + ...
                       Fo*(1 - (1/(2*n)))*T(n-1);
        end
    end

    T = T_new;
    time = time + dt;

    % Store data for plotting
    times(end+1) = time;
    center_T(end+1) = T(1);
    surface_T(end+1) = T(N);

    % Snapshots
    if isempty(snapshot_30s) && time ≥ 30
```

```
        snapshot_30s = T;
    end
    if isempty(snapshot_2min) && time ≥ 120
        snapshot_2min = T;
    end
end

% Final snapshot
snapshot_final = T;

% === Plot centerline and surface temperature ===
figure;
plot(times, center_T, 'r-', 'LineWidth', 2); hold on;
plot(times, surface_T, 'b--', 'LineWidth', 2);
xlabel('Time [s]');
ylabel('Temperature [ C ]');
legend('Centerline', 'Surface');
title('Hot Dog Cooking: Centerline & Surface Temperature vs Time - Numerical ...
    Method');
grid on;

% === Plot temperature distribution at 3 times ===
figure;
plot(r, snapshot_30s, 'k-', 'LineWidth', 2); hold on;
plot(r, snapshot_2min, 'b-', 'LineWidth', 2);
plot(r, snapshot_final, 'r-', 'LineWidth', 2);
xlabel('Radial Position r [m]');
ylabel('Temperature [ C ]');
legend('30 sec', '2 min', sprintf('Final (%.1f s)', time));
title('Temperature Distribution in Hot Dog at Different Times - Numerical ...
    Method');
grid on;
```