

The Effect of Burner Size on Pan Temperature Distribution

Kyle Webster
(Dated: February 2025)

The steady state solutions of a reduced model of the heat flow in an idealized frying pan were numerically computed for various bottom temperature distributions, representing heating from burners of different sizes. It was found that a burner radius of approximately 70% of the pan radius produced the most even usable heating for a 12" cast iron pan. The steady-state temperature model resulted in a 2D Laplacian with nonhomogenous Robin boundary conditions. It was discretized using central finite differences, resulting in a sparse linear system, solved using the SciPy routine `spsolve`. The applicability and usefulness of the model could be improved by acquiring more accurate burner temperature distributions for various sizes and gas flow rates and applying the model to more types of pans.

Keywords: Laplace Equation; Numerical PDEs; Cooking

I. INTRODUCTION

Although cooking is ubiquitous in the world, the expertise and skills of chefs is highly varied due to its complex and time-consuming nature. One challenge that chefs face is selecting the appropriate heat source and cook surface combination for specific techniques; while some cooking techniques require a central hot area, others may be better suited to more uniform distributions. For example, cooking a good omelette demands control over a pan's temperature distribution to prevent uneven cooking. Household stovetops have varying sizes and types of burners, producing different temperature distributions along the bottom of pots and pans. This choice of heat source and therefore temperature distribution to cook with presents an additional challenge to be navigated in order for chefs to cook as they intend. Therefore, characterizing the effects of these temperature distributions can aid many chefs in deciding what heat source to use when.

This study aims to characterize a cast iron pan's steady-state temperature distribution's dependence on burner radius by using the numerical solution of a PDE that models the heat conduction in an idealized pan. Cast iron pans are common in household and restaurant kitchens, and have a reputation for more evenly dispersing heat. Analyzing a model for the temperature distributions in these pans is therefore practical and potentially useful to many chefs. To make this characterization of the steady-state temperature distribution, a model of the physical process at hand, heat transfer, must be developed and then solved.

II. MODEL

The model consists of 3 main components: 1. the geometry of the pan 2. heat conduction in the pan 3. heat transfer at the boundary of the pan. It should be noted that radiative heat transfer is not included as it introduced nonlinearity.

Firstly, the shape of the cast iron pan was idealized into a radially symmetric cylinder with uniform material properties. This is done as heating elements and pans are very often both circular, leading to radial symmetry in the temperature distribution unless asymmetries in the cooling of the pan are present. As for dimensions, a 30cm diameter and 0.5cm thick cylinder was used, corresponding to a 12" pan.

Secondly, and perhaps most simply, conductive heat transfer in the pan was modelled by the heat equation. Since the domain is a radially symmetric cylinder, a cylindrical formulation of the heat equation was used and which when looking for steady-state solutions results in a Laplace equation as the time derivative is zero.

Finally, the convective heat transfer at the boundary of the pan was modelled by the heat-transfer coefficient model, producing nonhomogenous Robin boundary conditions for three of the four sides to equip the Laplace equation with. The final boundary is at the center of the pan and was given a no flux condition as there is no material at negative radius for the heat to be transferred to. This choice also conveniently removes the singularity at $r = 0$ introduced by using cylindrical coordinates.

Furthermore, the external temperature for the boundary conditions must be suitably chosen. For the cook surface, room temperature was chosen as it was assumed the heat from the pan and burner does not influence the temperature of the room, and that there is sufficient airflow to replace the air near the pan before it significantly changes temperature. For the bottom surface, the temperature distribution was modelled using two intervals of normal distributions; one for the center to flame tip, and the other for the flame tip to the pan radius. The two normal distributions were tuned to approximate the temperature distribution of a liquified petroleum gas (LPG) stove in Widodo et al.[1] with

a central temperature of approximately 700°C and a flame temperature of 1100°C . The normal distributions were parameterized in burner and pan radius so that any size burner could be used with any size pan. Improving the temperature distributions for other burner sizes would improve the accuracy of the model. For the outer surface, the bottom temperature distribution at the pan radius was used as the hot gas from the burner wraps around the pan as it rises.

With the high-level model described, the derivation of the numerical model that is solved is given in this section. Note that vectors are in bold-face and matrices are underlined. The notation for some quantities important to the model are given ahead of their use here.

Position, Time	\mathbf{x}, t
Cylindrical Coordinates	r, φ, z
Temperature Distribution	$T(\mathbf{x}, t)$
Bottom Temp. Dist.	$T_b(r)$
Steady-state Temp. Dist.	$T(r, z)$
Temp. Difference on Boundary	$\Delta T(\mathbf{x}, t)$
Pan interior	Ω
Pan boundary	$\partial\Omega$
Pan radius and height	r_p, z_p
Density	ρ
Heat Capacity	c
Heat conductivity	k
Heat-transfer coefficient	h
Heat flux vector	\mathbf{q}
Normal vector	\mathbf{n}
Gradient	∇
Laplacian	∇^2

A. Physics Background

To investigate the temperature distribution of a frying pan above a burner, it is important to model the transfer of heat between a gas and a solid (convective heat transfer) and within the solid (conductive heat transfer). A common model of convective heat transfer between a flow and a boundary is the Heat-transfer coefficient model[2], giving the heat transfer from a fluid to a wall as

$$\mathbf{q} \cdot \mathbf{n} = h(T_f - T_w)$$

where T_f and T_w are the temperatures of the fluid and wall respectively.

The other important model of heat transfer to consider is conduction. This is modelled by the heat equation[3], which can be derived by applying the conservation of energy to a stationary and uniform volume of mass.

$$\rho c \frac{\partial T}{\partial t} - k \nabla^2 T = a$$

where a is the heat production density. The second term comes from applying the law of heat conduction, $\mathbf{q} = -k\nabla T$, to the heat flux through the surface of a test volume; note that this law will be useful in determining the boundary conditions of the PDE model later. In the case of the pan atop a burner, $a = 0$ as the pan does not produce heat itself.

B. Model Derivation

The model is derived from the principle of conservation of energy. Since the pan is a solid, only the conduction of heat internally and the heat flux on the boundary of the pan is incorporated – radiative heat transfer is not included. A 3-dimensional model is initially considered and reduced to finding the steady-state temperature distribution in a cross-section of a radially averaged pan. The full model is the following:

$$\rho c \frac{\partial T}{\partial t}(\mathbf{x}, t) = k \nabla^2 T(\mathbf{x}, t) \quad \text{in } \Omega \tag{1a}$$

$$\mathbf{q} \cdot \mathbf{n} = h \Delta T(\mathbf{x}, t) \quad \text{on } \partial\Omega \tag{1b}$$

with the Laplacian in cylindrical coordinates,

$$\nabla^2 T(\mathbf{x}, t) = \frac{\partial^2 T}{\partial r^2}(\mathbf{x}, t) + \frac{1}{r} \frac{\partial T}{\partial r}(\mathbf{x}, t) + \frac{1}{r^2} \frac{\partial^2 T}{\partial \varphi^2}(\mathbf{x}, t) + \frac{\partial^2 T}{\partial z^2}(\mathbf{x}, t)$$

where Ω is the interior of the pan, defined by $\Omega = [0, r_p] \times [0, 2\pi] \times [0, z_p]$, and $\partial\Omega$ is the boundary of Ω .

To reduce this model $T(\mathbf{x}, t) \rightarrow T(r, z, t)$ is imposed – interpreting $T(r, z, t)$ as the average temperature over each ring of the pan – and only the steady-state, $\frac{\partial T}{\partial t} = 0$, is considered. With $T(r, z)$, the radially averaged steady-state solution to (1a), and $\mathbf{q} = -k\nabla T$, the reduced model is the following:

$$-\frac{k}{\rho c} \left(\frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial z^2} \right) = 0 \quad \text{in } \Omega \quad (2a)$$

$$-\frac{k}{h} \frac{\partial T}{\partial z}(r, z_p) + T(r, z_p) = T_{\text{air}} \quad (2b)$$

$$\frac{k}{h} \frac{\partial T}{\partial z}(r, 0) + T(r, 0) = T_b(r) \quad (2c)$$

$$-\frac{k}{h} \frac{\partial T}{\partial r}(r_p, z) + T(r_p, z) = T_b(r_p) \quad (2d)$$

$$k \frac{\partial T}{\partial r}(0, z) = 0 \quad (2e)$$

Note that the boundary condition at $r = 0$ is due to the radial symmetry imposed on the problem. Also note that the key dependence to study is $T(r, z)$'s dependence on $T_b(r)$. Summarily, (2a) describes what the steady-state temperature distribution satisfies on the pan's interior and the remaining equations, (2b–2e), are for the boundary conditions in the order of top, bottom, exterior, and central boundaries – the central boundary only exists due to taking radial symmetry. The next step is to perform nondimensionalization to get a well-scaled model for discretization.

C. Nondimensionalization

To nondimensionalize the steady-state model, scalings are introduced for each variable. That is,

$$T(r, z) = [T]T^*(r^*, z^*), \quad r = [r]r^*, \quad z = [z]z^*$$

Furthermore, defining the dimensionless parameters $\alpha = \frac{[r]}{[z]}$ and $\beta = \frac{-k}{h[z]}$, and setting $[r] = r_p$, $[z] = z_p$, and $[T] = T_{\text{air}}$, the following nondimensionalized model was derived:

$$\frac{\partial^2 T^*}{\partial r^{*2}} + \frac{1}{r^*} \frac{\partial T^*}{\partial r^*} + \alpha^2 \frac{\partial^2 T^*}{\partial z^{*2}} = 0 \quad \text{in } \Omega^* \quad (3a)$$

$$\beta \frac{\partial T^*}{\partial z^*}(r^*, 1) + T^*(r^*, 1) = \hat{T} \quad (3b)$$

$$-\beta \frac{\partial T^*}{\partial z^*}(r^*, 0) + T^*(r^*, 0) = \hat{T}_b(r^*) \quad (3c)$$

$$\frac{\beta}{\alpha} \frac{\partial T^*}{\partial r^*}(1, z^*) + T^*(1, z^*) = \hat{T}_b(1) \quad (3d)$$

$$\frac{\partial T^*}{\partial r^*}(0, z^*) = 0 \quad (3e)$$

With $\hat{T}_b(r^*) = T_b([r]r^*)/[T]$, the nondimensionalized burner heat distribution, and $\Omega^* = [0, 1] \times [0, 1]$, the nondimensionalized domain. Note that the superscript * will be dropped in later sections to declutter the equations. The model given above was solved numerically for different burner temperature distributions $T_b(r)$.

D. Burner Temperature Distribution

The temperature distribution of the air/gas mixture at the bottom surface of the pan was parameterized in r , r_b , and r_p by dividing the domain into $[0, r_b]$, and $[r_b, r_p]$. The piecewise definition of the burner temperature as a function of radius using two normal distributions that was used is the following:

$$T_b(r, r_b, r_p) = \begin{cases} T_{center} + (T_{flame} - T_{center})e^{-5\left(\frac{r}{r_b}-1\right)^2} & r \in [0, r_b] \\ T_{exterior} + (T_{flame} - T_{exterior})e^{-10\left(\frac{r}{r_b}-1\right)^2} & r \in [r_b, r_p] \end{cases} \quad (4)$$

with $T_{center} = 970 - 500\frac{r_b}{r_p}$ [K], $T_{flame} = 1370$ [K], and $T_{exterior} = 370$ [K].

III. SOLUTION METHOD

This section explains the details of the solution method. At a high level, r and z are discretized to form a regular grid T takes values on, central finite differences are applied to discretize all derivatives, the boundary conditions are applied to determine the value of ghost points, and the resulting linear system is solved by a state-of-the-art method for sparse linear systems to get the steady state temperature at each grid point.

A. Discretization

The discretization of Ω creates a regular grid with spacings d_r and d_z to form an $n \times m$ grid. That is, $d_r = 1/n$ and $d_z = 1/m$. Additionally, ghost points outside the domain are introduced to impose the boundary conditions on the interior points. Fig (1) provides a visualization of the 2 dimensional discretization; the points outside of Ω representing the ghost points. The following shorthand notation is used:

$$\begin{aligned} r_i &= \frac{i}{n}, & z_j &= \frac{j}{m}, & T(r_i, z_j) &= T_{i,j} \\ i &= 1, \dots, n, & j &= 1, \dots, m \end{aligned} \quad (5)$$

Both first and second order derivates were discretized using central finite differences. For a general function approximated on a regular grid of spacing d , the central finite difference schemes read as the following:

$$f'(x_i) \approx \frac{f_{i+1} - f_{i-1}}{2d}, \quad f''(x_i) \approx \frac{f_{i-1} - 2f_i + f_{i+1}}{d^2}$$

Applying the discretizations above to (3a–3e) results in the discrete model given in terms of indices.

PDE:

$$\frac{1}{d_r^2}(T_{i+1,j} - 2T_{i,j} + T_{i-1,j}) + \frac{1}{r_i} \frac{1}{2d_r}(T_{i+1,j} - T_{i-1,j}) + \frac{\alpha^2}{d_z^2}(T_{i,j+1} - 2T_{i,j} + T_{i,j-1}) = 0 \quad (6a)$$

Boundary Conditions:

$$\frac{\beta}{d_z}(T_{i,m+1} - T_{i,m}) + T_{i,m} = \hat{T} \quad (6b)$$

$$-\frac{\beta}{d_z}(T_{i,1} - T_{i,0}) + T_{i,1} = \hat{T}_b(r_i) \quad (6c)$$

$$\frac{\beta}{\alpha d_r}(T_{n+1,j} - T_{n,j}) + T_{n,j} = \hat{T}_b(1) \quad (6d)$$

$$T_{1,j} - T_{0,j} = 0 \quad (6e)$$

Note that at the boundary conditions have been discretized with the forward finite difference scheme instead of central finite differences. This model can be solved by arranging the equations into a single matrix equation of the form $\underline{A}\vec{x} = \vec{b}$ and then applying an appropriate technique to solve a sparse linear system. Also note that the notation for the vectors in the linear system are different to those used in the model derivation as they do not have a physical correspondence.

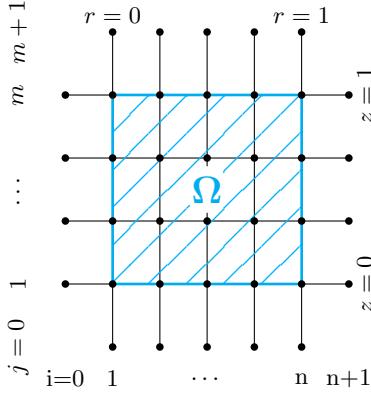


FIG. 1. Visualization of the discretization $T(r, z) \rightarrow T_{i,j}$ on Ω including ghost points for $n = 5$, $m = 4$. Note that each \bullet is a $T_{i,j}$. The grid is not uniform as the scale of features in the solution may not be similar in the r and z directions.

B. Forming the Linear System

To form the linear system which can be solved by state-of-the-art techniques, the temperature at each point on the grid is rearranged into a vector \vec{T} , collecting the interior points in increasing r , then increasing z so that $\vec{T}_{nj+i} = T_{i,j}$. This gives rise to a block structure of m rows of m blocks of size $n \times n$. The boundary conditions are formed by replacing the ghost point values by expressions in terms of the interior points. The system of equations then takes the following form:

$$\underline{\mathbf{A}}\vec{T} = \begin{bmatrix} \underline{\mathbf{L}}^{bc} & \underline{\mathbf{D}}_{zz}^\pm & & \\ \underline{\mathbf{D}}_{zz}^\pm & \underline{\mathbf{L}}^{in} & \underline{\mathbf{D}}_{zz}^\pm & \\ & \ddots & \ddots & \ddots \\ & & \underline{\mathbf{D}}_{zz}^\pm & \underline{\mathbf{L}}^{in} & \underline{\mathbf{D}}_{zz}^\pm \\ & & & \underline{\mathbf{D}}_{zz}^\pm & \underline{\mathbf{L}}^{bc} \end{bmatrix} \vec{T} = \vec{b} = \begin{bmatrix} \vec{f}_{bot} \\ \vec{f} \\ \vdots \\ \vec{f} \\ \vec{f}_{top} \end{bmatrix} \quad (7a)$$

$$\underline{\mathbf{L}}^{loc} = \underline{\mathbf{D}}_{rr} + \underline{\text{diag}}(1/r)\underline{\mathbf{D}}_r + \underline{\mathbf{D}}_{zz}^{loc} \quad (7b)$$

with the following numerical values:

$$\underline{\mathbf{D}}_{rr} = \frac{1}{d_r^2} \begin{bmatrix} -1 & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & \ddots \\ & & 1 & -2 & 1 \\ & & & -1 & \frac{\alpha d_r}{\beta} - 1 \end{bmatrix}, \quad \underline{\mathbf{D}}_r = \frac{1}{2d_r} \begin{bmatrix} 0 & 0 & & \\ -1 & 0 & 1 & \\ & \ddots & \ddots & \ddots \\ & & -1 & 0 & \frac{\alpha}{\beta r_n} \end{bmatrix} \quad (7c)$$

$$\underline{\mathbf{D}}_{zz}^\pm = \frac{\alpha^2}{d_z^2} \mathbf{I}_n, \quad \underline{\mathbf{D}}_{zz}^{in} = -2 \frac{\alpha^2}{d_z^2} \mathbf{I}_n, \quad \underline{\mathbf{D}}_{zz}^{bc} = \left(\frac{d_z}{\beta} - 1 \right) \frac{\alpha^2}{d_z^2} \mathbf{I}_n \quad (7d)$$

$$\vec{f} = \left[0 \ 0 \ \cdots \ 0 \ \left(\frac{1}{d_r} + \frac{1}{r_n} \right) \frac{\alpha}{\beta} \hat{T}_b(1) \right]^T, \quad \vec{f}_{bot} = \vec{f} + \frac{\alpha^2}{d_z \beta} \vec{T}_b, \quad \vec{f}_{top} = \vec{f} + \frac{\alpha^2}{d_z \beta} \hat{T} \quad (7e)$$

where $[\vec{T}_b]_i = \hat{T}_b(r_i)$. Now, with $\underline{\mathbf{A}}$ and \vec{b} , $T_{i,j}$ can be solved for by an appropriate sparse linear system solver. The implementation here used the `scipy` routine `spsolve` to solve the sparse linear system. See Davis et al.[4] for the algorithm details.

IV. RESULTS

The effect of burner size on the steady-state temperature distribution of a 12" cast iron pan, with a bottom thickness of 0.5cm, was investigated. The cast iron pan was modelled by using the following parameter values:

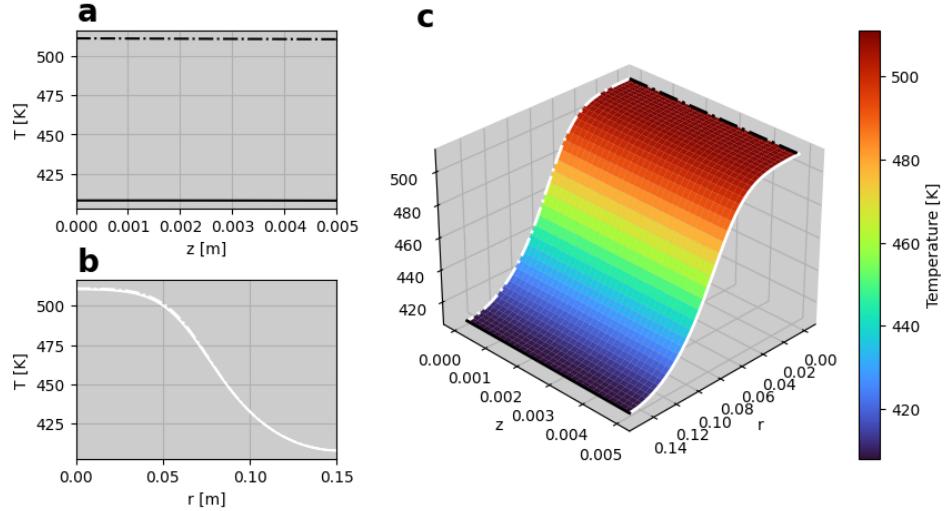


FIG. 2. **a:** Internal temperature distribution at the center and edge of the pan. **b:** Radial temperature distribution at the top and bottom of the pan. **c:** Radially symmetric temperature distribution of the entire pan. Note the correspondence of lines plotted in a and b to those in c.

$r_p = 0.15$, $z_p = 0.005$, $k_{\text{cast iron}} = 40$, $h_{\text{cast iron}} = 20$. With $r_b = 0.4r_p$ and the parameters above, Fig (2) was generated from the solution given by the numerical solver, the code for which is available in appendix A. Note that the burner temperature distribution code is also in appendix A. Figure (2) shows that the pan gets hottest at the center and that the temperature from the bottom to the top of the pan is nearly constant. There is furthermore a difference of 103 Kelvin between the inside and outside of the pan. This demonstrates that burner choice can induce large differences in temperatures throughout the pan and therefore indicates that characterizing the pan surface temperature distribution's dependence on the burner temperature distribution is useful in helping chefs choose what size burner to use.

To characterize the dependence on the burner temperature distribution the model was run for 100 different burner radii with the same parameters for the 12" cast iron pan. Fig (3) visualizes this dependence. Most important to note is that the most even heating in the pan occurs when the burner radius is approximately 70% of the pan radius, and that generally pan temperature increases with burner radius.

Additionally, the smallest radius burner also provides a mostly even pan surface temperature distribution. However, it is not hot enough to be used in most cooking applications as it is only approximately 80°C. Moreover, a burner of one hundredth a 12" pan is extremely unlikely. That is, the smallest burner radius can be ignored as a potential choice for generating even pan heating.

V. DISCUSSION

The investigation into the nature of a 12" cast iron pan's surface temperature distribution with respect to the burner it is atop revealed that there is an optimal burner radius to achieve the most even heating in the pan. However, the conclusions are heavily dependent on the input burner temperature distributions. Given that the model used for the burner temperature distribution was based on data from a single LPG stove, there are significant improvements that can be made. Specifically, models based on data from other burners of multiples sizes, perhaps common household stovetops, or models that allow for the burner strength to be adjusted would improve the applicability of the results. Moreover, revising the pan bottom boundary condition to better reflect the air gas mixture flow conditions at the boundary could improve the model's accuracy. Further work could include models of the bottom temperature distribution for other types of heat sources, such as electric or induction stoves to improve applicability.

An additional change that could be made is to consider only a 1 dimensional model which assumes constant temperature in the thickness of the pan. Fig (2) demonstrates that for the 12" cast iron pan geometry and thermal properties, the steady-state temperature is nearly constant across its thickness, meaning the 1 dimensional model would be a good approximation. However, including thickness in the model may be essential for different pan geometries and materials. For example, thicker bottoms on copper or stainless steel pots are prevalent configurations worth considering that may need the thickness in the model.

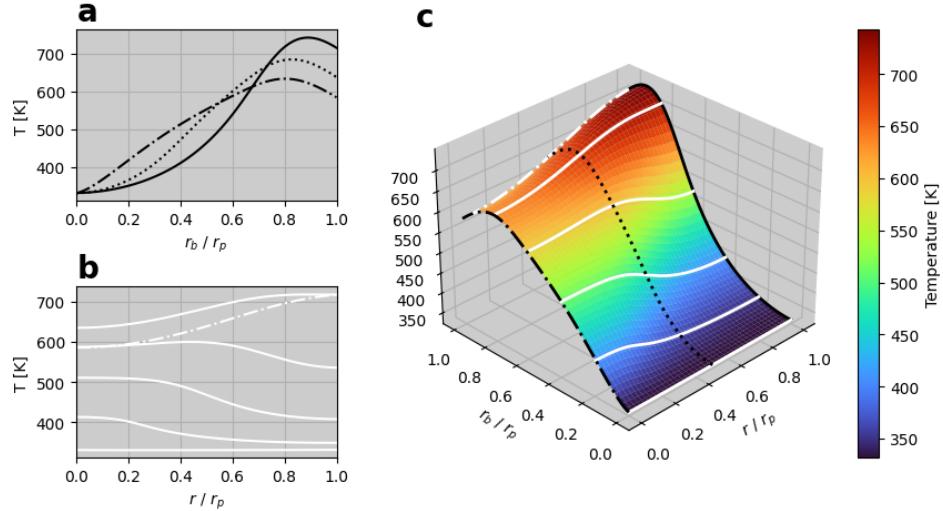


FIG. 3. **a:** Temperature profiles for changing burner radius at $r = 0$, $0.5r_p$, and r_p . Close proximity of all three lines indicates even pan heating. **b:** Temperature profiles at constant burner radius. Straighter lines indicate more even heating. **c:** Visualization of the full radial temperature distribution's dependence on burner radius. Note that the most even temperature distributions are for the smallest radius burner and for a burner approximately 70% of the pan radius.

Further work is required to improve the accuracy of the boundary conditions. Using free convection heat transfer coefficients for all boundaries except the center certainly introduces inaccuracies. Moreover, cooking situations could be incorporated by modifying the top boundary to have food cooking at certain locations to apply the model to specific cooking techniques. Additional further work could be done on many other pans and/or pots that the existing model could be used for to investigate if there is a trend in optimal burning size for material properties or thickness.

A final suggestion for further work could be finding the pan that gives the most even heating at its cook surface over a range of burner sizes and strengths. This may be useful in informing the design choices, such as material or thickness, or purchase decisions regarding future pans. The model given here, which finds the steady-state temperature distribution in pans of both arbitrary cylindrical geometry and material properties for arbitrary burner temperature distributions, is a useful tool for these purposes.

Appendix A

Pan Model Code

```
from typing import Callable
import numpy as np
from scipy.sparse import diags_array, block_array
from scipy.sparse.linalg import spsolve

def pan_model(
    r_pan : float,
    z_pan : float,
    k : float,
    h : float,
    T_exterior : float,
    T_burner : Callable[[float],float],
    n : int,
    m : int
) -> tuple[np.ndarray]:
    """Solves for the steady-state temperature distribution in a pan
```

Given the physical properties of a pan and its base's temperature distribution due to a burner, solves a model of heat transfer to find the steady-state temperature distribution of the pan. Performs a nondimensionalization of the input parameters, discretization into an $n+1$ by $m+1$ grid, formation of a linear system, and finally a sparse linear solve.

Args:

`r_pan` (float): radius of the pan in meters
`z_pan` (float): thickness of the pan in meters
`k` (float): thermal conductivity of the pan in
 Watts per meter Kelvin
`h` (float): heat transfer coefficient of the pan in
 Watts per meter² Kelvin
`T_exterior` (float): temperature on the sides and top of the pan
`T_burner` (Function(float) -> float): pan base temperature as a
 function of radius
`n` (int): grid points in the r direction
`m` (int): grid points in the z direction

Returns:

(`R`, `Z`, `T`) - the tuple of 3 np.ndarrays containing the grid point
 locations in `r`, `z`, and the steady-state temperature at each point.

`"""`

Step 1: Nondimensionalization

Scalings

```
r_scale = r_pan
z_scale = z_pan
T_scale = T_exterior
```

Apply Scalings

```
T_hat_burner = lambda r: T_burner(r_scale * r) / T_scale
T_hat = T_exterior / T_scale
```

Dimensionless Parameters

```
beta = -k / (h*z_scale)
alpha = r_scale / z_scale
```

Step 2: Discretization

`r`, `z` on [0, 1] x [0, 1] - the interior `r` and `z` values

```
d_r = 1/n
d_z = 1/m
r = np.linspace(0, 1, n)
z = np.linspace(0, 1, m)
```

`T` with ghost points - the pan starts at room temp
`T_0` = np.ones(n*m) * `T_hat`

Discretized bottom temperature distribution
`T_b` = np.array([`T_hat_burner`(ri) for ri in r])

Step 3: Forming Sparse Linear System

```
I_n = diags_array(np.ones(n), format="csr")
```

Radial Derivatives as matrices

```
D_rr = (1 / d_r**2) * diags_array([1, -2, 1], offsets=[-1, 0, 1],
                                    shape=(n,n), format="csr")
D_r = (1 / (2*d_r)) * diags_array([-1, 0, 1], offsets=[-1, 0, 1],
                                    shape=(n,n), format="csr")
```

diag(1/r)

At $r=0$, the boundary condition forces (partial of T in r)=0
so (partial of T in r)/ r -> 0 at $r=0$ means we can ignore the
first component of $r[i]$ if it is zero before inverting.

```
r_copy = np.copy(r)
if (r[0] == 0):
    r_copy[0] = 1
diag_inverse_r = diags_array((1/r_copy), offsets=0,
                               shape=(n,n), format="csr")
```

Z Derivatives

```
D_zz_in = (-2 * alpha**2 / d_z**2) * I_n
D_zz_pm = (alpha**2 / d_z**2) * I_n
```

```

# Forcing
f = np.zeros(n)

# Impose Boundary conditions
# Inner boundary
D_rr[0,0] = -1 / d_r**2
D_r[0,1] = 0

# Outer boundary
D_rr[-1,-1] = ((alpha*d_r / beta) - 1) / d_r**2
D_r[-1,-2] = 0
D_r[-1,-1] = alpha / (beta*r[-1])
f[-1] = (1/d_r + 1/r[-1]) * (alpha / beta) * T_b[-1]

# Top boundary
# Use these blocks when creating the top row instead
D_zz_bc = (alpha**2 / d_z**2)*(-1+d_z/beta) * I_n
f_top = f + (alpha**2 / (d_z*beta)) * T_hat

# Bottom boundary
# Use D_zz_bc for bottom row as well
f_bot = f + (alpha**2 / (d_z*beta)) * T_b

# Entire System
# interior row main diagonal
L_in = D_rr + diag_inverse_r @ D_r + D_zz_in
L_bc = D_rr + diag_inverse_r @ D_r + D_zz_bc

# List of blocks
blocks = []
for j in range(m):
    if (j == 0):
        blocks.append([L_bc, D_zz_pm] + [None]*(m-2))
    elif (j == m-1):
        blocks.append([None]*(m-2) + [D_zz_pm, L_bc])
    else:
        blocks.append([None]*(j-1) + [D_zz_pm, L_in, D_zz_pm] + [None]*(m-j-2))
A = block_array(blocks, format="csr")
b = np.concatenate((f_bot, np.tile(f,m-2), f_top))

# Step 4: Solving the linear system
T = spsolve(A, b)

# Step 5: Prepare Output
R, Z = np.meshgrid(r*r_scale, z*z_scale)
T_steady_state = T.reshape(m,n)*T_scale

return (R, Z, T_steady_state)

```

Burner Temperature Distribution Code

```
def T_b_distribution(r : float, r_b : float, r_p : float) -> float:
    """Burner Radial Temperature Distribution
```

Gives the temperature at the bottom of a pan from a burner of radius r_b at radius r for a pan of radius r_p .

Args:

*r (float): radius to get temperature at
 r_b (float): burner radius
 r_p (float): pan radius*

Returns:

the temperature at radius r due to a burner of size r_b

```
normal_dist = lambda r, mu, var: np.exp(-(r-mu)**2 / var)
T_central = 970.0 - 500.0 * (r_b/r_p)
T_fire = 1370.0
T_ext = 290.0 + 80.0
```

```
if r < r_b:  
    return T_central + (T_fire-T_central)*normal_dist(r/r_b, 1, 2e-1)  
else:  
    return T_ext + (T_fire-T_ext)*normal_dist(r/r_b, 1, 1e-1)
```

- [1] A. Widodo, Sudarno, S. Soeparman, and S. Wahyudi, The effect of finned heat reflector materials and diameters on the efficiency and temperature distribution of liquefied petroleum gas stove, Results in Engineering **16**, 100658 (2022).
- [2] M. Favre-Marinet, S. Tardu, and W. O. Library, *Convective heat transfer: solved problems*, 1st ed. (ISTE, London;Hoboken, NJ;; 2009;2010;2013;).
- [3] *Mathematics 461: Projects in Applied Mathematics* (UBC, 2025).
- [4] T. A. Davis, J. R. Gilbert, S. I. Larimore, and E. G. Ng, Algorithm 836: Colamd, a column approximate minimum degree ordering algorithm, ACM Trans. Math. Softw. **30**, 377–380 (2004).