

Alex Gonzalez, Ben Harker, and Kyle Wahlberg

Abstract

We started this project with a dream, a dream of meat on the grill and summer nights. That dream turned into us sitting, staring at a computer screen, longing for our code to compile. Our initial idea was to model how different kinds of meat cook on different kinds of surfaces, using the numerical methods that we learned in class about FTCS, Heat Advection, and some kind of Graphical User Interface.

Procedure and Problem Statement

For our project we wanted to model the grilling and searing of different meats using diffusion equations for conduction and convection. We began this project with the FTCS method. The meat starts at a fixed temperature (42°F) as if it had just been taken out of the fridge. Heat is then applied to the meat at a constant temperature (though differing in method) for as long as it takes the meat to reach optimal “doneness”. All of this showed on our GUI that the user could interact with to select different options for cooking, meat type, temperature, and other variables. The GUI is shown below in Figure :

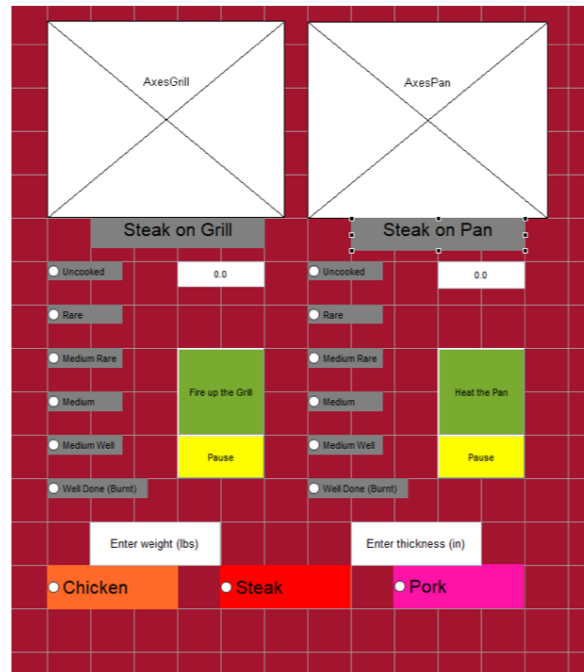


Figure 1: MATLAB Project Graphical User Interface

```

clear all;
specificHeatCapacity = 0.56; % beef btu/(lb F)
density = .033; %beef lbs/ci
thermalcond = .185 / 12; % beef btu/(hr F ft) / 12 in/ft
thermaldiff = thermalcond/(density*specificHeatCapacity);

tempInitial = 42; % degrees F
rare = 120;
medRare = 130;
med = 140;
medWell = 150;
burnt = 160;

weight = input('How much meat (lbs) are you cooking? ');
thick = input('How thick (in) is your steak? ');
done = menu('How would you like your steak cooked?', 'Rare', 'Medium', 'Rare', 'Medium', 'Medium Well', 'Burnt');
method = menu('How are you cooking?', 'Cast Iron', 'Grill');

% For FTCS Numerical Method
N = 100;
L = sqrt((weight / density) / thick);
hx = L/N; % Space grid size
hy = thick/(2*N);
x = (0:N)*hx; % Space coordinate
time=0;
tau = 0.00001;

```

Figure 2: MATLAB code screenshot 1 (see appendix for more)

To get us a starting point with some of the math that goes into the heat energy transfer to meats, we found two papers, one a master's thesis and the other a doctoral. From these papers we got the equations for conduction, convection, and radiation. Radiation is quite complicated to model, and less important than conduction or convection, so we did not attempt to model it. Conduction and convection seemed straightforward enough, however we ran into some issues.

We realized that the steady state conductivity equations we received from the paper are exactly that, steady state. In our case this means that the heat transfer wouldn't stop until the meat reached the same temperature as the grill, which is quite beyond the ideal medium rare. So after much heartache, we returned to the FTCS method. Additionally, the FTCS method is time dependent, as is the cooking of meat, which lines up quite nicely.

To use a lot of these equations, we needed some general information about meat. Below are some of the constants and variables we used in our program:

$$\begin{aligned}
 K &= 0.185 \frac{\text{BTU}}{\text{hr ft}^\circ\text{F}} \\
 C &= 0.56 \frac{\text{BTU}}{\text{lb}^\circ\text{F}} \\
 \rho &= 0.033 \frac{\text{lb}}{\text{ci}} \\
 \text{So... } \mathbf{k} &= \frac{k}{c \rho} = 0.834 \frac{\text{in}^2}{\text{hr}}
 \end{aligned}$$

We also found a table that gave us some information about different kinds of meats we would be grilling (Figure ??):

```

if method==1
    T(:,1)=450;
else
    T(:,1)=400;
    for i=1:4;
        T(round(i*N/4))=450;
    end;
end;

while T(:,N/2)< CookTemp
    T(ii,ji) = T(ii,ji) + thermaldiff*tau*( ((T(im,ji)-2*T(ii,ji)+T(ip,ji))/hx^2) +
    ((T(ii,jm)-2*T(ii,ji)+T(ii,jp))/hy^2) );
    T(:,2)=T(:,1);
    T(1,:)=T(2,:);
    T(N+1,:)=T(N,:);
    T(:,N+1)=T(:,N);

    plot(x,T(:,N/2),'k',[0,L],[rare,rare],'r',[0,L],[medRare,medRare],'r',[0,L],
    [med,med],'y',[0,L],[medWell,medWell],'y',[0,L],[burnt,burnt],'k');
    xlim([-1,L+1]); ylim([40,burnt]);
    xlabel('Steak (in)'); ylabel('Temperature (F)');
    pause(0.000001);
end;

```

Figure 3: MATLAB code screenshot 1 (see appendix for more)

Results

We had the user input weight of meat, thickness, desired doneness, and then they could select pan or grill. For the pan, we had a constant bottom temperature of 450 °F, and insulated boundaries for the other three sides. On the grill, we had four grates at 450 °F, and the rest at 400°F. This is based off general information that often air temps in grills are less than the actual fire or grate temperature. We also assumed insulated boundaries for the sides and top of the grill . Once all the data was entered, we used the 2D FTCS method to calculate the next temperature given its surroundings. See the appendix for samples of code. We used N=100 steps, but this could easily be changed for faster computation or more accuracy. The plot then showed the temperature at the halfway thickness point rising, stopping at the desired temperature. At this point in real life, we would flip the meat and repeat for the same amount of time of the other side. We also included a timer to show how long the meat would take to cook. These times were fairly accurate, however definitely idealized somewhat given we assumed no heat loss to surrounding air, constant temperature and density, and ignored more complex methods of heat transfer.

Appendix

Food	Method	Temperature	Time*
Brats	Grill	450-500	5-10 min./side
Brisket	Slow-cook	225-240-	6-10 hours
Burgers	Grill	450-500	7-10 min./side
Chicken pieces	Grill	375-425	15-20 min./side
Chicken - whole	Roast	275-325	1.5-2.5 hours
Fish	Grill	375-425	7-20 min./side
Hot Dogs	Grill	450-500	5-7 Min./side
Jerky	Smoke	150-170	4-6 hours
Pizza	Bake	400-450	14-18 minutes
Pork Butts/Picnics	Slow-cook	225-240	8-16 hours
Ribs	Roast	200-250	4-8 hours
Roast Beef	Roast	250-300	1.5-3.5 hours
Salmon - smoked	Smoke	150-175	6-10 hours
Steaks	Grill	450-500	6-10 min./side

Figure 4: MATLAB Project Graphical User Interface