Ps 4 Problem 1

Adam Guerin

October 2, 2023

Abstract

The most naive implementation of integration is with the the trapezoid rule or a riemann sum. There are much better computational methods with much better implementations already embedded in different public available Python packages. Here I describe implementing one to integrate the heat capacity of a chunk of aluminum.

1 Introduction

The following is the equation for the heat capacity of a solid at temperature T:

$$C_v = 9V\rho k_B(\frac{T}{\theta_D})^3 \int_0^{\frac{\theta_D}{T}} \frac{x^4 e^x}{(e^x-1)^2}\, dx \tag{1}$$

Where V is the volume, ρ is the density of the material, kb_B is the Boltzmann constant,and θ_D is the materials Debye temperature. In our case a 1000 cm^3 block of aluminum of density $6.022 * 10^{28} m^{-3}$ and Debye temperature 428 K.

2 Methods

You can use a direct gaussian quadrature but I just used scipy for this problem. I used quadrature in problem 3 of this homework directly so show the difference in using hermite polynomials and using legendre polynomials to solve the problem. Just make a function that returns this integral and plug it into the scipy fixed quadrature program with 50 points specified as shown in the code (making sure to not include constants in the integral). After doing that I plotted the specific heat for varying temperatures.

3 Results

Look at plots below. The temperature follows the trend we expect. Gaussian quadrature was just okay and required quite a few points to be accurate. I specifically chose a low temperature of 1.5 kelving as my test temperature to see how quadrature performed as the integral begins to blow up at your temperature approaches zero.

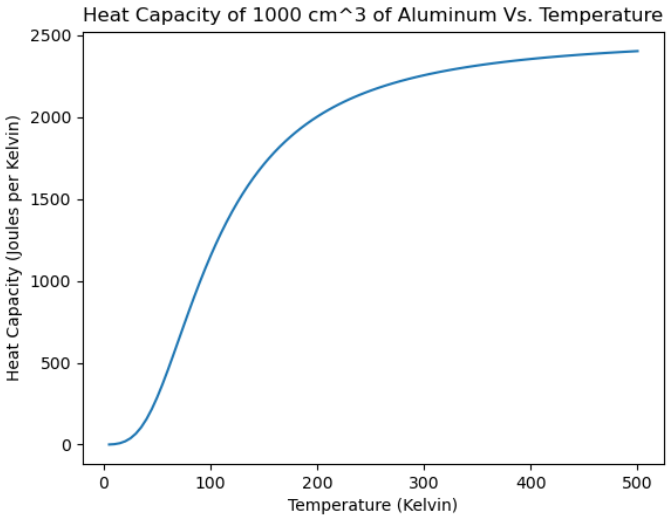

Figure 1: An expected heat capacity curve. Nothing too interesting, it levels off at higher temperature as the material becomes unable to hold onto more energy easily.

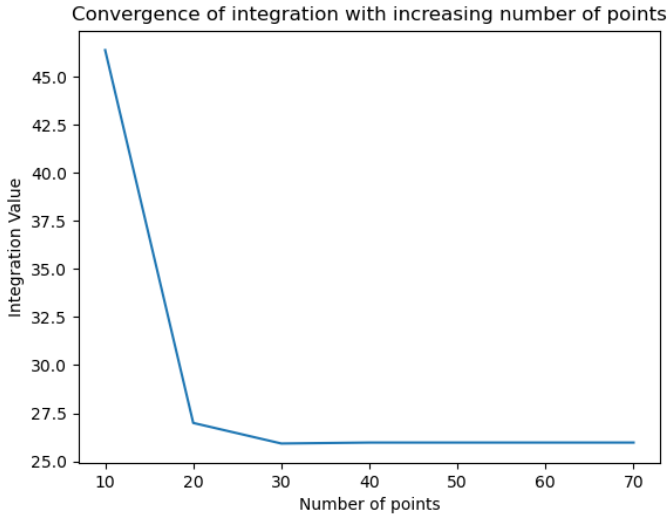

Figure 2: Gaussian quadrature isn't very good for this problem requiring 30 points before leveling off on an answer and accuracy

4 Discussion

Scipy is fine as a general source if you want to integrate something and especially of you know what the answer should be, but you should be more cautious and choose your integration method carefully if you don't know the answer or how well behaved your integral is. 30 points is kind of a lot. I show this more explicitly in problem 3 of this hw set.