Hot Plates

or Radiative Heat Transfer Across Parallel Plates in Series and It's Application to Packed Beds

Radiative heat transfer im packed beds has long been a nagging interest of mine. Through the years, I've modeled some pretty hot fixed bed reactors - north of 600C, even. At those temperatures, I assume that radiative heat from the solids in the bed is significant, but I have never accounted for it explicitly. I justify this oversite because, as far as I can tell, everyone else ignores it too.

As luck would have it, while looking into decarbonization technology options, I became interested in high temperature thermal batteries for energy storage to accomodate intermittent energy sources like wind or solar. These technologies can store heat at temperatures as hot as 2000K. To model these near flame temperature systems, I decided I would have to run bed radiative behavior to ground.

This article was a step in that journey. I call it Hot Plates.

I hope you enjoy it.

Start Simple

I'll build this analysis from a simple systems of parallel radiating plates. Figure 1 illustrates that system with a hot plate at one end and a cold plate at the other end. I fix the hot plate temperature at 2000K and the cold plate temperature at 400K.

As I'm sure we all recall from school days…

The radiation from the hot and cold plate is

$$\sigma\epsilon T_h^4 \qquad and \qquad \sigma\epsilon T_c^4$$

And so the net heat transfer is

$$q_{rad} = \sigma\epsilon F(T_h^4 - T_c^4)$$

We can make our lives easier for now by assuming the plates are blackbodies so $\epsilon = 1$, and if the distance between plates is small relative to the height, then the view factor, F, is also 1.

$$\epsilon = 1 \qquad F \approx 1$$

Then

$$q_{rad} = 5.67 * 10^-8(2000^4 - 400^4) = 9.06 * 10^5 \frac{W}{m^2}$$

Radiative heat transfer between two black body plates with an assumed view factor of 1

$$q_{rad} = \sigma \epsilon F (T_h^4 - T_c^4)$$

$$\epsilon = F = 1$$

$$q_{rad} = 5.67 * 10^{-8}(2000^4 - 400^4) = 9.06 * 10^5 \frac{W}{m^2}$$

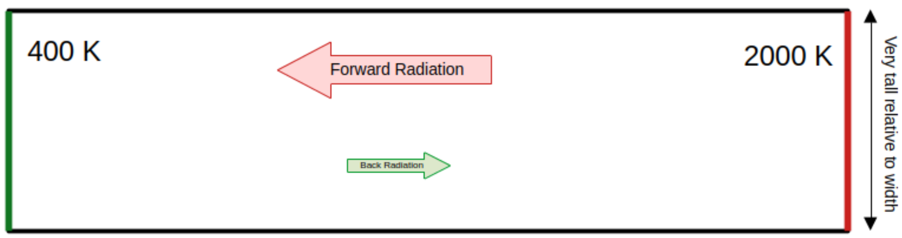


Figure 1: *A simple two plate radiative heat transfer system. We can simplify by making the plates black bodies and the system much taller than wide, so $\epsilon = F = 1$.*

Take Small Steps

Figure 2 is the same system, but this time we have added an intermediate plate. We will hold the hot and cold plates at 2000k and 400k and find the intermediate plate temperature when the system is at steady state.

The algebra isn't too bad. At steady state, at every plate, the flux arriving equals the leaving, so

$$\sigma(T_h^4 - T_m^4) = \sigma(T_m^4 - T_c^4)$$

then rearranging

$$T_m^4 = \frac{T_h^4 + T_c^4}{2}$$

And

$$T_m = 1682.5K$$

$$q_{rad} = 4.53 * 10^5 \frac{w}{m^2}$$

and we have cut the forward heat flux in half.

Radiative heat transfer between two black body plates with 1 parallel plate inserted

$$q_{rad} = \sigma(T_h^4 - T_m^4) = \sigma(T_m^4 - T_c^4)$$

$$T_m^4 = \frac{T_h^4 + Tc^4}{2}$$

$$T_m = 1682.5K$$

$$q_{rad} = 4.53 * 10^5 \frac{W}{m^2}$$

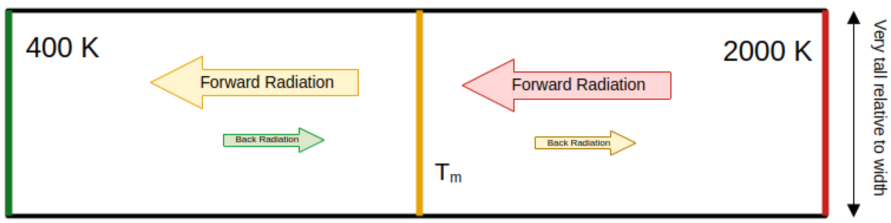


Figure 2: *The same system as Figure 1, but now there is an intermediate plate. We will solve for the intermediate plate temperature and also discover that the flux is reduce by* $\frac{1}{2}$.

As the astute reader may anticipate, we will now add a second plate. That system is shown in Figure 3.

At steady state

$$\sigma(T_h^4 - T_{warm}^4) = \sigma(T_{warm}^4 - T_{tepid}^4) = \sigma(T_{tepid}^4 - T_c^4)$$

Sparing the reader the algebra

$$T_{warm} = 1807.5K \qquad T_{tepic} = 1520.9K$$

$$q_{rad} = 3.02 * 10^5 \frac{W}{m^2}$$

and we have cut the forward heat flux by a factor of 3.

Radiative heat transfer between two black body plates with 2 intermediate plates

$$q_{rad} = \sigma(T_h^4 - T_{wrm}^4) = \sigma(T_{wrm}^4 - T_{tepid}^4) = \sigma(T_{tepid}^4 - T_c^4)$$

$$T_{wrm} = 1807.5K \qquad T_{tepid} = 1520.9K$$

$$q_{rad} = 3.02 * 10^5 \frac{W}{m^2}$$

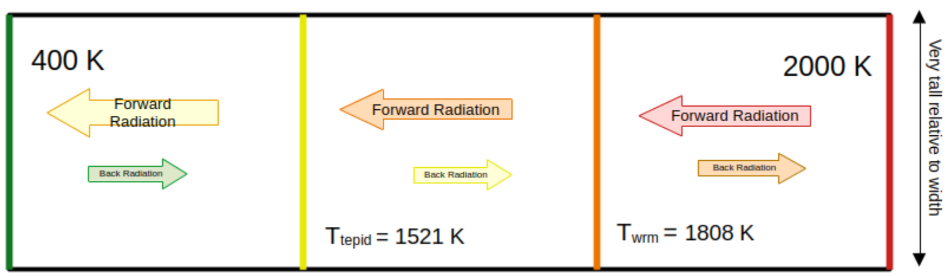


Figure 3: *The same system as Figure 1, but now there are two intermediate plates. We will solve for the intermediate plate temperatures and also discover that the flux is reduced by a factor of 3.*

Until We Reach a Comfortable Pace

Well, the jogging analogy might be bogging down, but I hope the reader can see how to extend our simple systems to arbitrarily large systems of N intermediate plates as shown in Figure 4.

At each intermediate plate, n_i, steady state requires that:

$$(T_{i+1}^4 - T_i^4) = (T_i^4 - T_{i-1}^4)$$

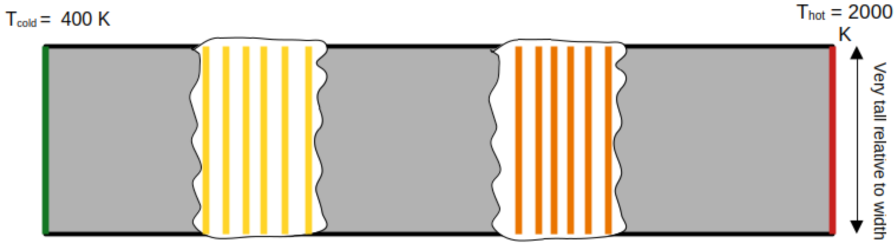
which we can expand to

$$\begin{bmatrix} -2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -2 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} -T_{hot} \\ 0 \\ 0 \\ \vdots \\ -T_{cold} \end{bmatrix}$$

The first and last equations are different than the interior equations because we know $T_h and T_c$. I"m sure the curious reader can verify that this system is correct.

Sparse Matrix Parallel Plates in Series Problem

$$\begin{bmatrix} -2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -2 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} -T_{hot} \\ 0 \\ 0 \\ \vdots \\ -T_{cold} \end{bmatrix}$$



$$at\ steady\ state\ (T_{i+1}^4 - T_i^4) = (T_i^4 - T_{i-1}^4)\ at\ each\ plate$$

Figure 4: *Now we have a large system of N intermediate plates. At steady state, the equations across the plates make a handy sparse matrix system. We will learn that the flux is inversely proportional to $N+1$, and $N+1$ is number of plate to plate intervals that we will call A.*

We can rewrite our math more compactly as matrix multiplication

$$\mathbf{At} = \mathbf{r}$$

Where **A** is our sparse matrix; **t** is the vector of unknown plate temperatures we need to find; **r** is zero everywhere except the first and last entries, the hot and cold plate temperatures

Then a Sprint to the Finish

Python code to solve the system just defined is attached to this article as an appendix. I encourage the reader to use it and explore system behavior on their own.

Figure 5 is a loglog plot of the forward heat flux against $N+1$. That relationship is linear and the equation of that line is shown on the plot. The astute reader may have anticipated from our simple systems that the forward flux is inversely proprtional the number of intervals between plates, and indeed it is so. I will call the number of intervals $A = N+1$ where N is the number of intermediate plates. Pressing on:

$$\log_{10}(flux) = -\log_{10}(I) + 5.96$$

After some algebra we learn that

$$flux * A = 9.06 * 10^5$$

And the loglog y-intercept of 5.96 ($q_{rad} = 10^{5.96}$) tells us the flux corresponding to $A = 1$, which matches our result from Figure 1, as it should. We can rearrange to better reflect the inverse proprtionality.

$$flux = \frac{9.06 * 10^5}{A}$$

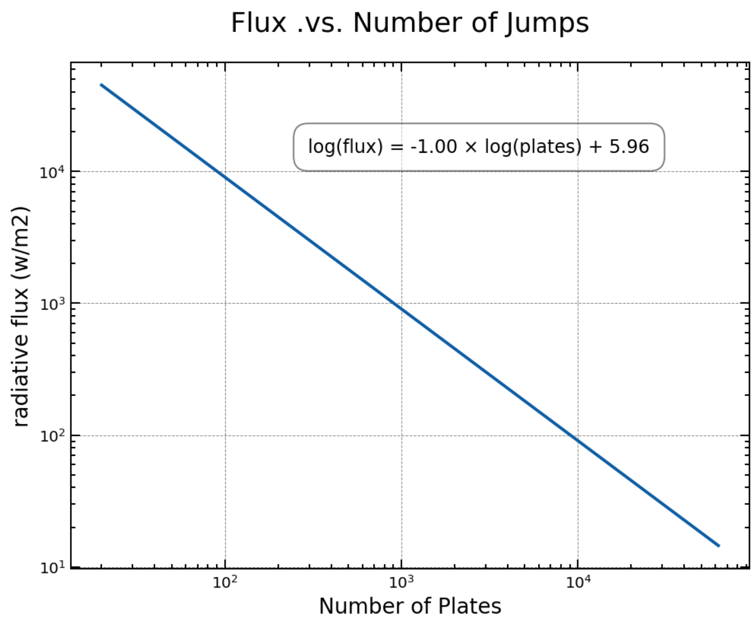


Figure 5: *A loglog plot of our large systems forward flux and the equation of the line is shown. The flux is inversely proportional to A, the number of intervals, consistent with our observation for the simpler systems.*

A comment about notation. In this article I am using N to represennt the number of intermediate plates. The total number of plates, N_{tot}, is N plus the hot plate and the cold plate, or $N + 2$. The number of intervals I will call A and it is equal to $N + 1$ or $N_{tot} - 1$.

Figure 6 is a plot of the temperature profiles, one atop the other for 60 cases with N ranging from 21 to 62,397. The temperature profile is invarient to the number of plates.

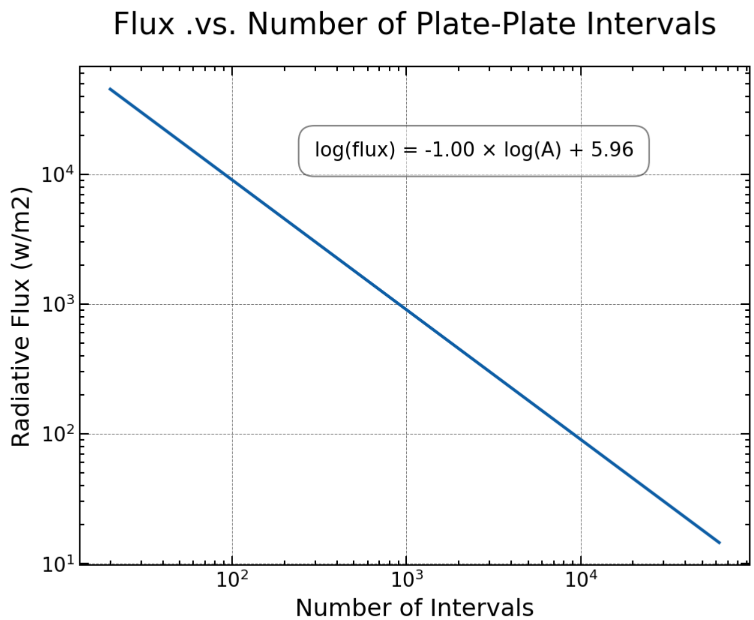


Figure 6: *The temperature profiles, for 60 cases with* $N = (21\,to\,62,397)$, *plotted, one atop the other. As is typcial, the lines are actually piecewise linear between the individual calculated points. Nevertheless, the profiles are indistinguishable from each other except at the lower right where the piecewise nature of the cases with the fewest number of plates are revealed.*

In Figure 7, I've plotted the profile of T^4 for the case with 62,397 plates. One need not be too awfully astute to notice that it is linear. So, $\frac{\partial T^4}{\partial n}$ is constant at steady state. But we can go further, and I hope the relationship below is clear to the reader.

$$\frac{dT}{dn} = \frac{T_{hot}^4 - T_{Cold}^4}{A}$$

Where A is the number of plate to plate intervals.

I want to take a moment to clarify some details regarding derivatives and sign conventions we will soon encounter.

The heat flux, q_{rad} will flow from high temperature to low temperature and so the radiative heat flux vector will have the opposite sign of the temperature gradient. This is the same as for conduction, except our driving force is T^4 instead of T.

I define $\frac{\partial}{\partial n}$ to be the derivative with respect to moving across one interval from one plate to the next, so $\frac{\partial}{\partial n} = \frac{\partial}{\partial a}$. I will use $\frac{\partial}{\partial n}$

I need to also define two related quantities, $\dot{n} = \frac{1}{\delta}$: δ is the distance between plates and $\dot{n}$ is number of plates per fixed distance, or plate density.

We can count plates by integrating δ.

$$\int_0^L \dot{n}\,dx = \dot{n}L + C = \frac{L}{\delta} + C$$

we can find C by noting that $\frac{L}{\delta}$ is the number of intervals in our system, $N+1$ and at $x=0$ there is one plate so $C=1$ and for the total number of plates, $N_{tot} = N+2$, as it should.

$$\int_0^L \dot{n}\,dx = N + 2$$

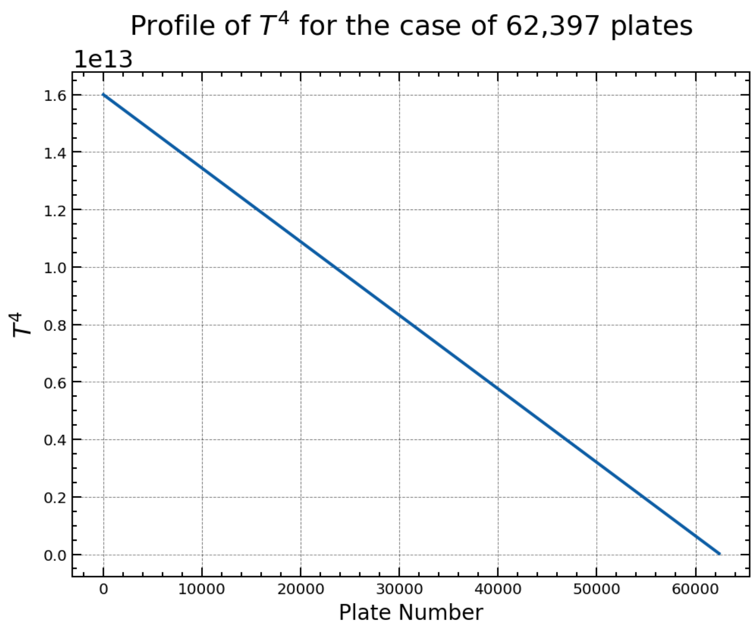


Figure 7: *The profile of T^4 for the case with the most plates. It is linear and we observe that $\frac{dT^4}{dn} = \frac{T_{hot}-T_{cold}}{A}$, which will lead us to a postulated consitutive equation for radiative flux that can we can apply in the context of a packed bed.*

Breaking the Tape

We now have a few key observations.

- The flux in our system is inversely proportional to A, the number of intervals between plates.
- The temperature profile is invarient to the number of plates
- At steady state, the profile of T^4 is linear and $-\frac{dT^4}{dn} = \frac{T_{hot}-T_{cold}}{A}$

The forward flux in our system is therefore:

$$q_{rad} = \sigma \frac{T_{hot}^4 - T_{cold}^4}{A} = -\sigma \frac{dT^4}{dn}$$

We can generalize to the case where the plates are not blackbodies and we need a view factor.

$$\boxed{q_{rad} = -\sigma \epsilon F \frac{dT^4}{dn}}$$

And, voila, we have a constitutive equation for radiative flux in a system of parallel plates, and the negative sign recognizes that the heat flux vector is opposite the temperature gradient.

Now to the Podium

It isn't much of a leap to apply these learnings to the case of radiative heat transfer in a packed bed.

Imagine a packed bed where the only physics at play is radiative heat transfer. Don't be bothered that this is unrealistic, we can readily add back the neglected physics - conduction, heat release or uptake via chemical reaction, or heat transfer to a working fluid moving through the bed, for example - and those additions do not affect our derivation.

Using our new found consitutive equation, the heat balance for our imaginary bed

$$C_a \Delta x \rho_{bed} C_p^s \frac{\partial T}{\partial t} = -\sigma \epsilon F S_a \left[\left. \frac{\partial T^4}{\partial n} \right|_{n=i} - \left. \frac{\partial T^4}{\partial n} \right|_{n=i+1} \right]$$

now we swap variables using

$$\frac{\partial T^4}{\partial n} = \frac{\partial T^4}{\partial x} \frac{dx}{dn}$$

and realizing that

$$\frac{dx}{dn} = \delta = \frac{1}{\dot{n}}$$

Where

$$\delta \equiv distance\ between\ plates$$

$$\dot{n} \equiv plates\ per\ fixed\ length$$

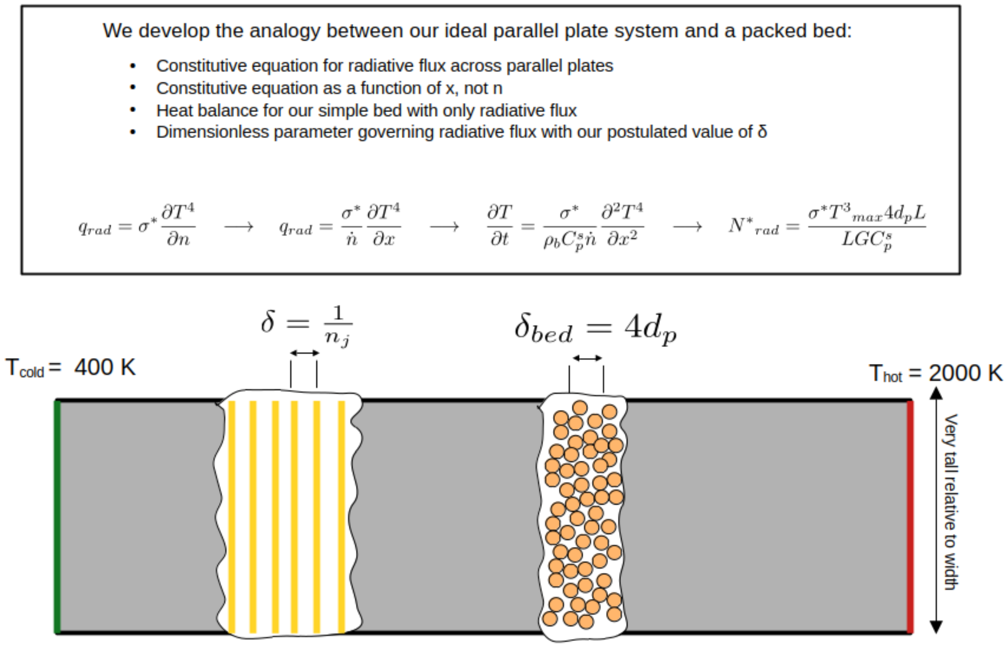


Figure 8: *An illustration of how we apply our math for parallel plate fluxes to flux behavior in packed beds. We swap independant variable so that temperature is $T(x)$, not $T(n)$, and we define δ as the maximum view distance along the bed axis.*

And we can rewrite our balance as

$$C_a \Delta x \rho_{bed} C_p^s \frac{\partial T}{\partial t} = \frac{-\sigma \epsilon F S_a}{\dot{n}} \left[\left. \frac{\partial T^4}{\partial x} \right|_x - \left. \frac{\partial T^4}{\partial x} \right|_{x+\Delta x} \right]$$

We have implicitly derived a modified constitutive equation for the case of radiative flux in a packed bed and swapped the discrete variables n or a, for the continuous spatial variable x.

$$q_{rad} = -\frac{\sigma^*}{\dot{n}} \frac{\partial T^4}{\partial x}$$

Now we do the usual: divide by Δx, take the limit, and rearrange the constants

$$\frac{\partial T}{\partial t} = \frac{\sigma^*}{\rho_b C_p^s \dot{n}} \frac{\partial^2 T^4}{\partial x^2}$$

And I have defined

$$\sigma^* = \sigma \epsilon F$$

This is a perfectly usable form that we can plug directly into a full featured packed bed model, and in a sense we are now done. But it is useful to work up a dimensionless version. We define dimensionless variables

$$x^* = \frac{x}{L} \qquad T^* = \frac{T}{T_{max}} \qquad t^* = \frac{t\dot{m}}{M}$$

Sparing the reader the algebra, we can rewrite our balance as

$$\frac{\partial T^*}{\partial t^*} = \frac{C_a \sigma^* M T_{max}^3}{\dot{m} C_a \rho_b C_p^s \dot{n} L^2} \frac{\partial^2 {T^*}^4}{\partial {x^*}^2}$$

but

$$L S_a \rho_b = M \quad and \quad \dot{n} L = A \quad and \quad \frac{\dot{m}}{S_a} = G$$

I have assumed the context of a thermal battery so $\dot{m}$ would be the flowrate of working fluid through the bed and M the total mass of solid in the bed and this is the natural dimensionless time for a thermal battery. In other contexts, a different dimensionless time may be appropriate

Substituting the dimensionless parameters

$$\boxed{\frac{\partial T^*}{\partial t^*} = \frac{\sigma^* T_{max}^3}{A G C_p^s} \frac{\partial^2 {T^*}^4}{\partial {x^*}^2}}$$

and

$$N_{rad} = \frac{\sigma^* T_{max}^3}{A G C_p^s} = \frac{\delta \sigma^* T_{max}^3}{L G C_p^s}$$

would be a dimensionless quantity of some import.

We still need to define what we mean by $\dot{n}$ or δ in a packed bed. Conceptually, there should be an analogy between our δ and HETP of a packed bed for distillation. I imagine that δ is the length of packed bed equivalent to one of the parallel plates in our simple radiative transfer models from earlier. So δ is a property of the bed material and the bed geometry.

We can use a little creativity to approximate δ for the case of a bed of packed spheres. Remember that radiative heat transfer depends, literally, on line of site - for a cold target to recieve radiative heat from a hot source, it must have a view of the source. Imagine our (very tiny) selves dropped into our packed bed, and then ask: how far downstream can we see? For a randomly packed bed of spheres I would postulate my view to be 3 or 4 particle diameters - I don't think I would be able to see a particle 5 diameters away.

Now, the reader is free to imagine the situation differently, but I would postulate that

$$\delta = \frac{1}{\dot{n}} = 4 d_p$$

Then we can modify our equation above by noting

$$A = L\dot{n} = \frac{L}{\delta} = \frac{L}{4d_p}$$

and I would postulate that N_{rad} for a randomly backed bed of spheres is

$$N^*{}_{rad} = \frac{4d_p\sigma^* T^3_{max}}{LGC^s_p}$$

And I'm sure the imaganitive reader can envision ways to estimate δ for other bed arrangements.

These learnings inform us about design strategy for packing structures in high temperature thermal batteries. We want to minimize radiative heat transfer by minimizing δ (or maximizing $\dot{n}$). But let's not get too far ahead of ourselves because we will have more to say about this as we develop a full featured high temperature thermal battery model. In any case…

I hope you enjoyed Hot Plates. And I hope you stay tuned for the rest.

Nomencature

$A \equiv$ Total number of plate to plate intervals
$a \equiv$ A single plate to plate interval
$C_a \equiv$ Cross sectional area of the bed, m^2
$C_p{}^s \equiv$ Solid heat capacity, $\frac{joule}{kgK}$
$d_p \equiv$ Particle Diameter, m
$F \equiv$ View Factor
$G \equiv$ Mass Flux, $\frac{kg}{m^2 s}$
$L \equiv$ Total System Length, m
$M \equiv$ Total mass of solid in the bed, kg
$\dot{m} \equiv$ Mass flow rate of working fluid through the bed, $\frac{kg}{s}$
$N \equiv$ Total number of intermediate plates (excludes the hot and cold plate)
$N_{tot} \equiv$ Total number of plates including the hot and cold plates
$\dot{n} \equiv$ Plates per fixed length, m^{-1}
$n \equiv$ A single plate $N_{rad} \equiv$ Dimensionless parameter characterizing radiative heat transfer
$N^*{}_{rad} \equiv$ Modified $N^*{}_{rad}$ using σ^*
$q_{rad} \equiv$ Radiative heat flux $\frac{W}{m^2}$
$T \equiv$ Temperature, K
$x \equiv$ Spatial parameter along the bed axis, m
$\delta \equiv$ Plate to plate interval distance, m
$\epsilon \equiv$ Solid Emmisivity
$\sigma \equiv$ Stephan Boltzman Constant, $5.67 * 10^{-8} \frac{W}{m^2 K^4}$
$\sigma^* \equiv$ Modifed σ where $\sigma^* = \sigma\epsilon F$

Python Code

```
#===============================================================
#----   importing all the usual stuff
#-------------------------------------------------------

import numpy as np
import scipy as sci
from scipy import optimize
import matplotlib.pyplot as plt
import scienceplots
plt.style.use(['science', 'notebook', 'grid'])
import scipy as sci
from scipy.sparse import diags
from scipy.sparse import csc_array
from scipy.sparse import csr_array
from scipy.sparse.linalg import spsolve


#================================================================
#--------  code
#=================================================================

T_hot = 2000
T_cold = 400
sigma = 5.67e-8
eps_f = 1.0
i_start = 20
i_num = 60
#i_end = 500500
flux = np.zeros(i_num)
x_pos = np.zeros(i_num)
#print(flux)
platenum = np.arange(i_start, i_start+i_num, 1) #not sure this is used



#========================================================
#-------------  Plot 1 ------------------------------
#========================================================

plt.figure()  # before starting a new plot

for count in range(i_start, i_start+i_num): #loops over requested iterations
```

```
    n_plates = round(5+1.15**count)   # lets number of plates grow exponentially
    print('n_plates =', n_plates)
    n = n_plates - 2 #number of equation is n_plates - 2
    #------------------------------------------------------------------------------------
    sub_C = 1
    main_C = -2
    sup_C = 1

    main = main_C * np.ones(n)
    sub =  sub_C * np.ones(n-1)
    sup = sup_C * np.ones(n-1)

    b = np.zeros(n)
    b[0] = -T_hot**4
    b[n-1] = -T_cold**4

    A = diags([sub, main, sup], offsets=[-1,0,1], shape = (n,n))
    A_csc = A.tocsc()

    soltn = spsolve(A_csc,b)  #solves the problem
    #-------------------------------------------------------------------------------------------
    flux[count - i_start] = (sigma*eps_f*(soltn[2]-soltn[3]))  #calculates the flux (should be
    x_pos[count - i_start] = n_plates-1 #number of jumps is n_plates - 1

    t_plot = np.concatenate([[2000], soltn**0.25, [400]])    #defines the array holding temper
    position = np.arange(len(soltn)+2)  #creates an array that counts from 0 to n + 2
    pos = position/(n_plates-1)
    #print(f"length of t_plot is: {len(t_plot)}")
    #print(f"length of position is:  {len(position)}")
    #print(f"number of plates is: {n_plates}")
    #print(f"pos of first plate is:  {pos[0]}, pos of last plate is {pos[-1]}")
    #position = list(range(1, lensoltn)+1))/n_plates
    plt.plot(pos, t_plot)
plt.title('Overlay of Temperature Profiles \n  for N = 21 to 62,367', fontsize=18, pad=20)
plt.xlabel('Dimensionless Position', fontsize=14)
plt.ylabel('Temp (K)', fontsize=14)
plt.tick_params(labelsize=10)

plt.show()

#===========================================================
#---- plot 2
#=============================================================

plt.figure()  # before starting a new plot
plt.plot(x_pos, flux)
plt.yscale('log')
```

```python
plt.xscale('log')
plt.title('Flux .vs. Number of Jumps', fontsize=18, pad=20)
plt.xlabel('Number of Plates', fontsize=14)
plt.ylabel('radiative flux (w/m2)', fontsize=14)
plt.tick_params(labelsize=10)

z = np.polyfit(np.log10(x_pos), np.log10(flux), 1)
p = np.poly1d(z)

#plt.plot(x_pos, 10**p(np.log10(x_pos)))
print(f"log10 of flux is:  {z[0]:.2f} times log10(plates) + {z[1]:.2f}")

equation = f'log(flux) = {z[0]:.2f} × log(plates) + {z[1]:.2f}'
plt.text(0.35, 0.85, equation, transform=plt.gca().transAxes,
         fontsize=12, verticalalignment='top',
         bbox=dict(boxstyle='round,pad=0.8', facecolor='white', alpha=0.5))

print(f"log10 of flux is:  {z[0]:.2f} times log10(plates) + {z[1]:.2f}")
plt.show()



#======================================================
#-----  Plot 3
#===========================================================

#print(soltn)
print(len(soltn))

plt.figure()  # before starting a new plot
xaxis = list(range(1, len(soltn)+1))
#print(xaxis)
print(len(xaxis))
temp = soltn
print(len(temp))
plt.plot(xaxis, temp)
plt.title(r'Profile of $T^4$ for the case of 62,397 plates', fontsize=18, pad=24)
plt.xlabel('Plate Number', fontsize=14)
plt.ylabel(r'$T^4$')
plt.tick_params(labelsize=10)
plt.show()
```