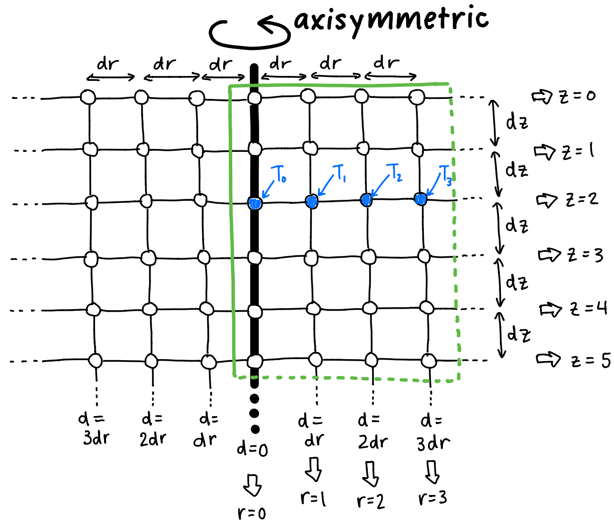
2D axisymmetric grid of temperature values:



dr, dz = the grid spacing, units of length

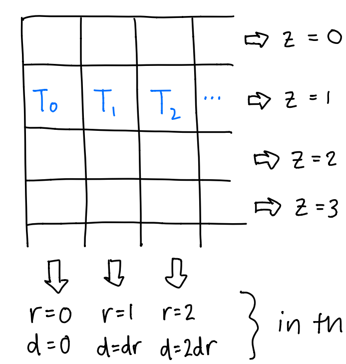
d = distance from the axis, units of length

r = <u>INDEX</u> of a grid point (starts at 0, b/c Python is zero-indexed)

T_r = temperature at a grid point, units of °C (only one row labeled)

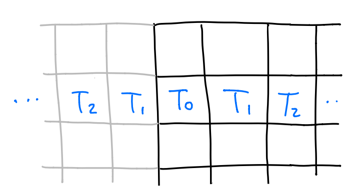
The array I operate on in the script is in the green box above; the leftmost column of values "is" the axis.

The array view:



} in this case, $d = r \cdot dr$

The "other side" of the axis starts w/ T_1:
(but only the array shown w/ black is what exists in the code)



Heat diffusion equation in cylindrical coordinates:

$$\frac{\partial u}{\partial t} = \alpha \left[\frac{1}{r}\frac{\partial u}{\partial r} + \frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} \right]$$

Each of these terms has a different finite difference approximation:

$$\frac{\partial u}{\partial r} : \frac{U[z,r+1] - U[z,r-1]}{2\,dr}$$

$$\frac{\partial^2 u}{\partial r^2} : \frac{U[z,r-1] - 2\cdot U[z,r] + U[z,r+1]}{dr^2}$$

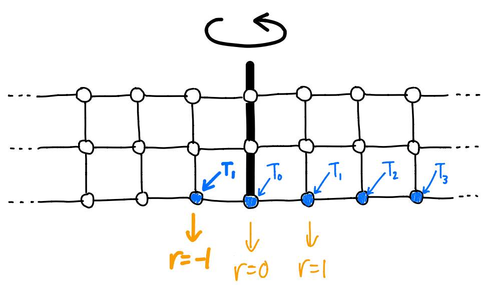
$$\frac{\partial^2 u}{\partial z^2} : \frac{U[z-1,r] - 2\cdot U[z,r] + U[z+1,r]}{dz^2}$$

Notation:

$U[z,r] = T_x$

array; row index; col index; value at that location in array

Circled terms need special consideration, since they effectively sample one to the left of the axis for **r=0** only.



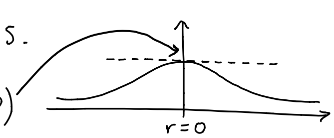
Because of axisymmetry, the temp value of $r=-1$ is the same as at $r=1$.

The $\partial u/\partial r$ term finite difference approximation becomes:

$$\frac{U[z,r+1] - U[z,r-1]}{2\,dr} \Rightarrow \frac{U[z,r+1] - U[z,r+1]}{2\,dr} = 0$$

Makes physical sense, since any change w.r.t. r across a symmetrical axis will be zero on the axis.

(derivative is zero)



The $\partial^2 u / \partial r^2$ term finite difference approximation becomes:

$$\frac{U[z,r-1] - 2\,U[z,r] + U[z,r+1]}{dr^2} = \frac{U[z,r+1] - 2\,U[z,r] + U[z,r+1]}{dr^2}$$

$$= \frac{2\,U[z,r+1] - 2\,U[z,r]}{dr^2}$$

These instances of $\partial u / \partial r$ and $\partial^2 u / \partial r^2$ are calculated only when the index $r=0$ (first column of the array).

Back to heat diffusion equation:

$$\frac{\partial u}{\partial t} = \alpha \left[\underbrace{\frac{1}{r}\frac{\partial u}{\partial r}} + \frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} \right]$$

At $r=0$, $1/r \to \infty$ but at $r=0$, $\partial u / \partial r = 0$, so term goes away.

And then back to the different finite difference approximations, now looking at $\partial^2 u / \partial z^2$:

$$\frac{\partial^2 u}{\partial z^2} : \frac{U[z-1,r] - 2 \cdot U[z,r] + U[z+1,r]}{dz^2}$$

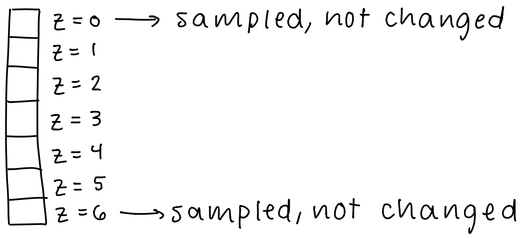
When $z=0$, have to deal w/ this term.

Solution: the loop over z values starts at $z=1$, not $z=0$.

The 0th row in the array ($z=0$) is sampled (term w/ orange arrow) but not changed w/ the heat flow function.

This is equivalent to the upper boundary being held at whatever temperature the cells in the first row are set to.

Want the same thing for the bottom boundary, so the ending z index will be total # of rows minus 1.



here: nrows = 7

needed iterations: 1, 2, 3, 4, 5

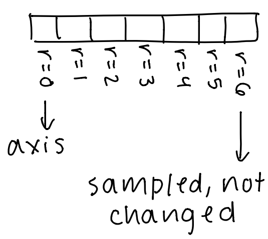
Python syntax:

```
for z in range(1,nrows-1)
```

[end of range is exclusive]

Loop syntax for rows (r indices):

- start w/ r=0 (need to capture the axis)
- end w/ next to last column (last column is sampled but not changed, like z behavior above)



here: ncols = 7

needed iterations: 0, 1, 2, 3, 4, 5

Python syntax:

```
for r in range(0,ncols-1)
```

[end of range is exclusive]

Now considering the energy calculation function:

$$Q = m \cdot c \cdot \Delta T = c \cdot \Delta T \cdot \rho \cdot V$$

Q → energy due to temp

c → specific heat, constant

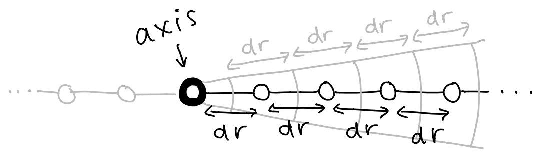
ΔT → temp of a cell relative to some reference temp

ρ → density, constant

V → cell volume, what I will calculate next here

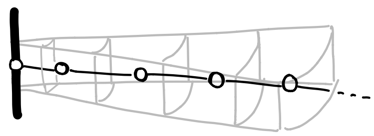
Cell volume increases as you move out along a row, since calculation is in cylindrical coordinates.

Looking down the axis/axis into the page:



Can divide the space around each grid point into cells

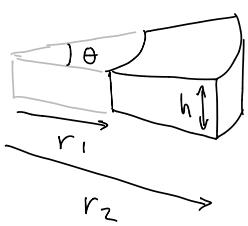
Oblique/side view:



The grid points are in the middle of the cells.

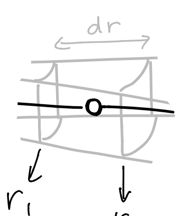
The volume of a general cell: $V = \frac{1}{2}\theta(r_2^2 - r_1^2)h$

where:



Call it $\frac{V}{\theta} = \frac{h}{2}(r_2^2 - r_1^2)$, since this code is doing everything on a 'per-angle' basis.

Converting cell volume equation into variables that the code uses. For one grid point not on the axis:



its index is r

its axis distance d is $r \cdot dr$

r_1 & r_2 (the radii of the inner & outer walls of the cell enclosing the grid point) are:

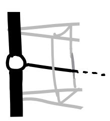
$$r_1 = d - \tfrac{1}{2}dr = r \cdot dr - \tfrac{1}{2}dr = dr(r - \tfrac{1}{2})$$

$$r_2 = d + \tfrac{1}{2}dr = r \cdot dr + \tfrac{1}{2}dr = dr(r + \tfrac{1}{2})$$

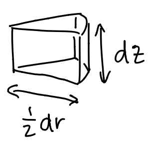
Plug these values into $V/\theta = h/2(r_2^2 - r_1^2)$ & simplify.

$$\frac{V}{\theta} = \frac{h}{2}\left(\left[dr\left(r+\tfrac{1}{2}\right)\right]^2 - \left[dr\left(r-\tfrac{1}{2}\right)^2\right]\right)$$
$$= \frac{h}{2}\left(dr^2\left(r+\tfrac{1}{2}\right)^2 - dr^2\left(r-\tfrac{1}{2}\right)^2\right)$$
$$= \frac{1}{2} dr^2 h\left(\left(r+\tfrac{1}{2}\right)^2 - \left(r-\tfrac{1}{2}\right)^2\right)$$
$$= \frac{1}{\cancel{2}} dr^2 h(\cancel{2}r)$$
$$= r\, dr^2\, h \qquad \text{(h is cell height, or dz here)}$$
$$= \underbrace{r}_{\text{index}}\ \underbrace{dr^2}_{\text{cell 'width'}}\ \underbrace{dz}_{\text{cell height}} \qquad \Rightarrow \text{units of volume} \checkmark$$

Considering the case of $r=0$, the axis:



$$\frac{V}{\theta} = \frac{h}{2}\left(r_2^2 - r_1^2\right) \qquad \begin{array}{l} r_1 \text{ here is zero} \\ r_2 \text{ here is } \tfrac{1}{2}dr \end{array}$$



$$\frac{V}{\theta} = \frac{h}{2}\left(\left(\tfrac{1}{2}dr\right)^2 - 0\right)$$
$$= \frac{h}{2} \cdot \frac{1}{4} dr^2$$
$$= \frac{1}{8} dr^2 dz$$

no index, because only for $r=0$

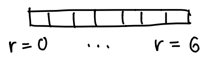
General volume eqn. w/ $r=0$: $\frac{V}{\theta} = (0)\, dr^2 dz = 0$

$\rightarrow$ Gives wrong answer, so need to treat $r=0$ separarately

(I think the axis is the only boundary that needs special treatment in this function)

(index z doesn't appear in volume eqn., so don't need to worry about when it's zero)

Check for statement iteration values:

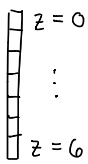


ncols = 7

need to see 0, 1, ..., 5, 6

Python syntax: for r in range(0, ncols)

[end of range is exclusive]



nrows = 7

need to see 0, 1, ..., 5, 6

Python syntax: for z in range(0, nrows)

[end of range is exclusive]