ACOUSTICS OF CONCERT HALLS

KIAN ORR, IVC

MENTOR: WILLIAM SPINELLA

BASICS OF ACOUSTICS

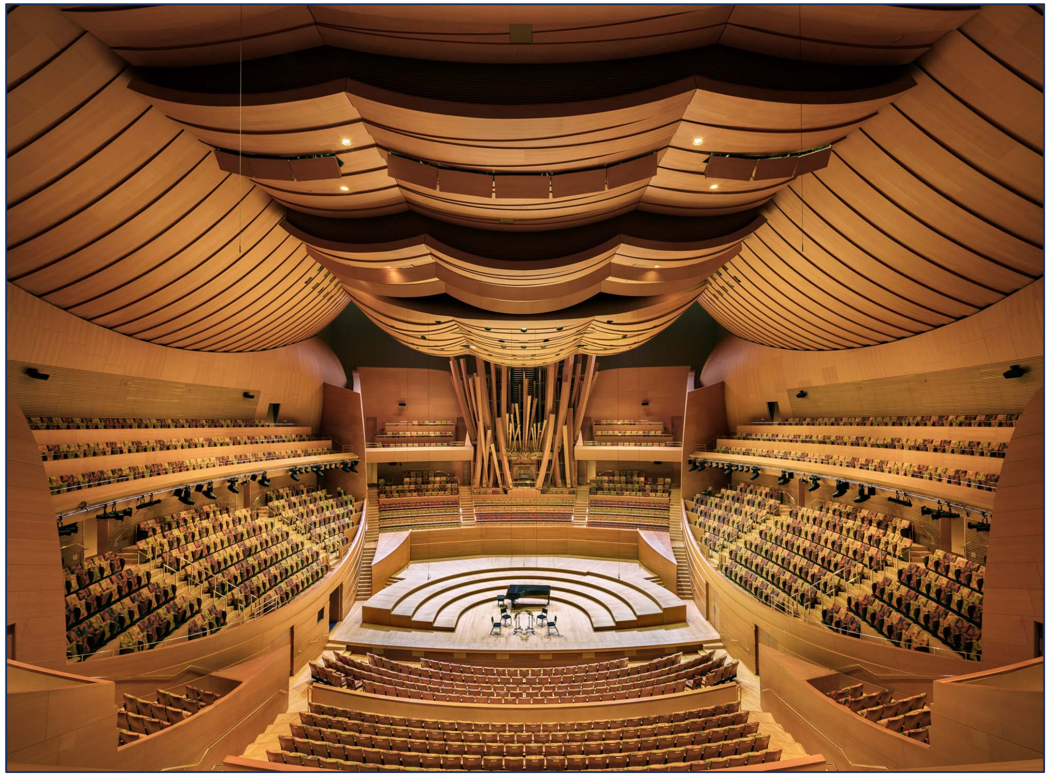
Sound waves

Frequency, amplitude, wavelength, phase difference, intensity

Acoustics

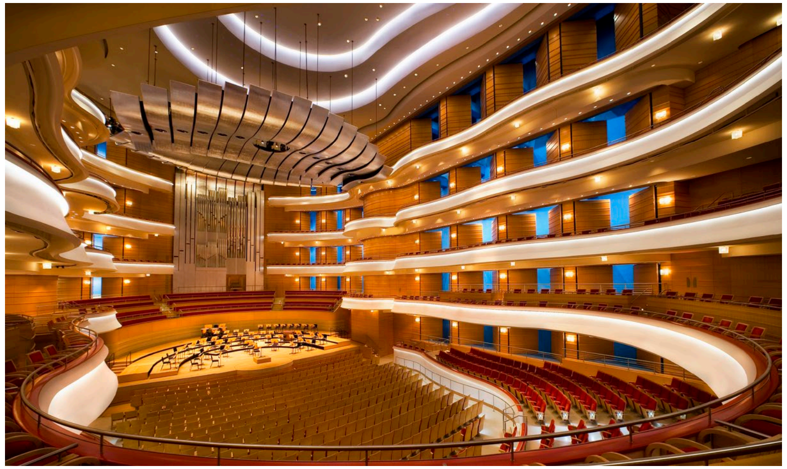
Reverberation, sound absorption, shapes of halls

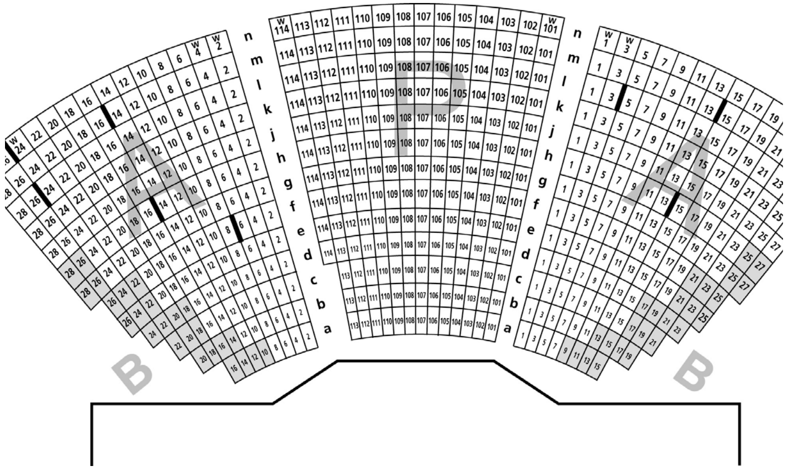
SURROUND HALLS



- Strength of sound is weaker because it is sent in a circle
- Lack of reverberation

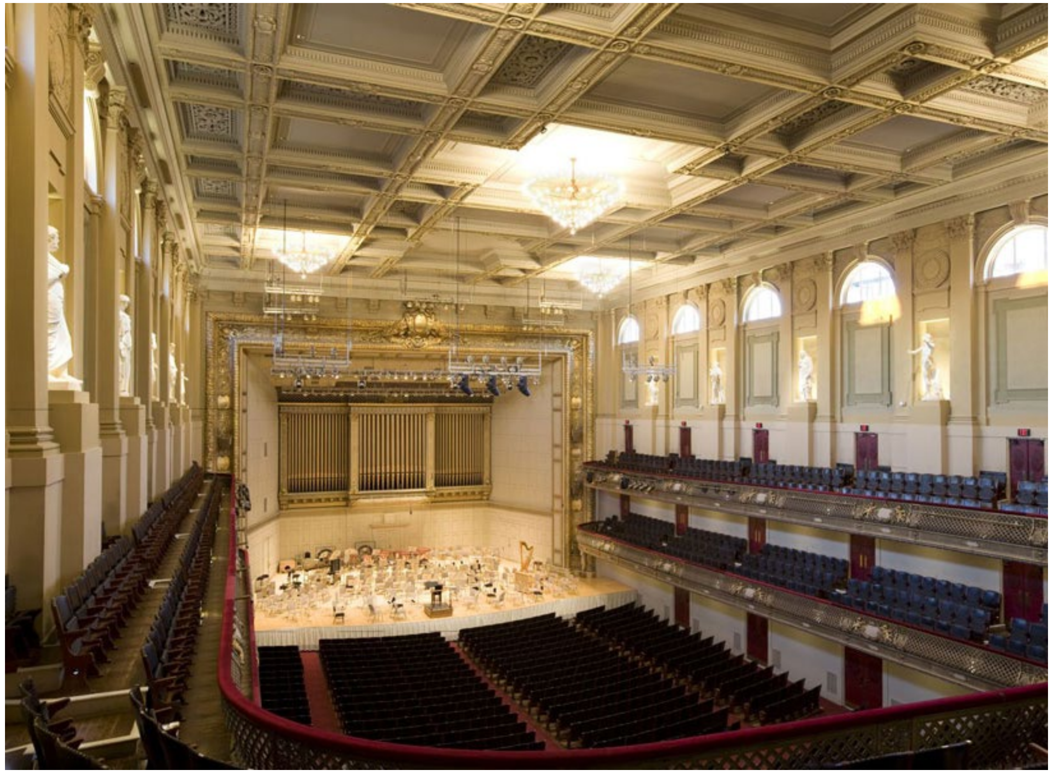
HORSESHOE HALLS



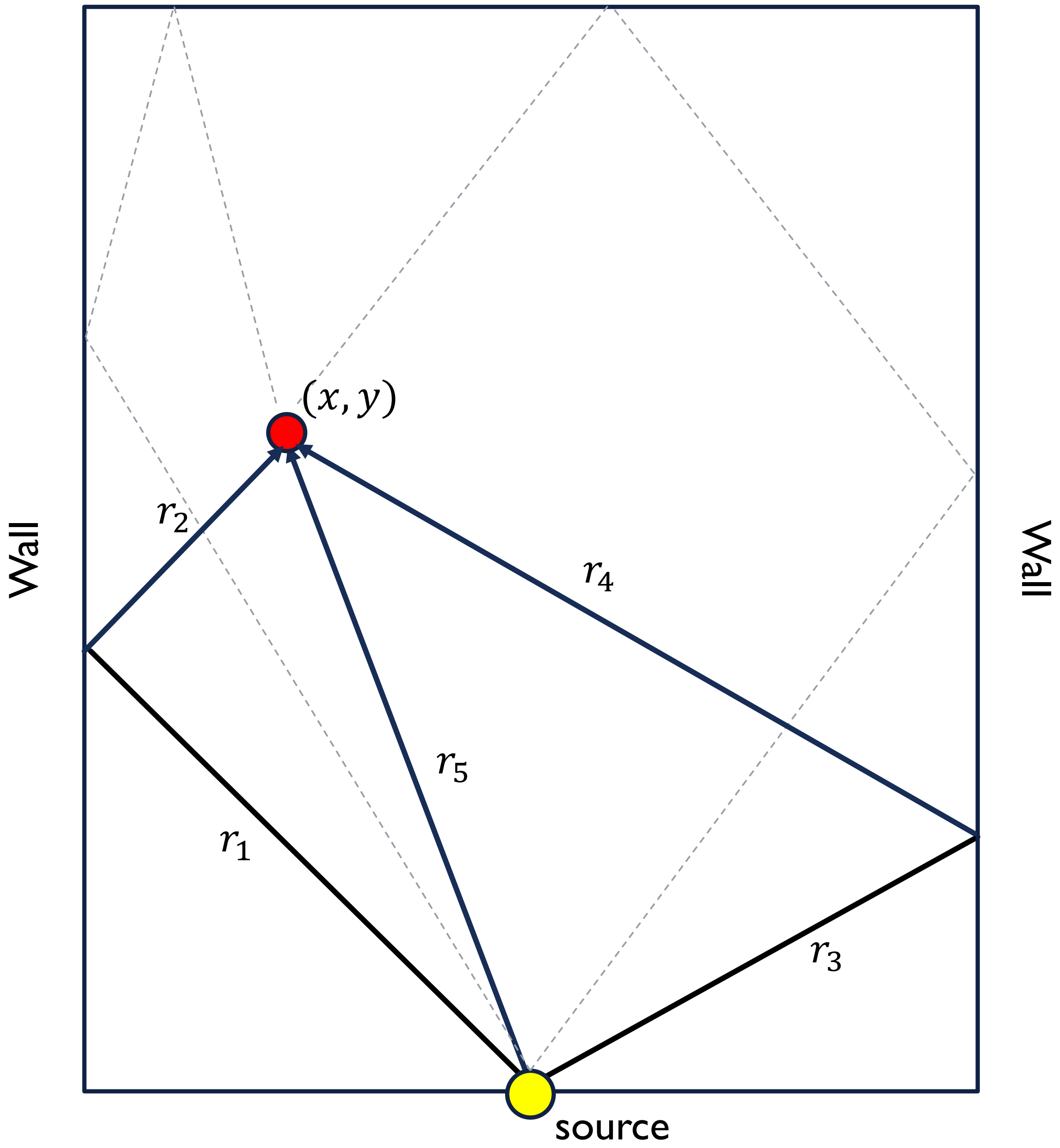


- Sidewalls have seating for the audience members which takes away the possibility of any reverberation occurring in that section
- Built for visual aspects and intimacy

SHOEBOX HALLS



- Shoebox halls are known to have the best acoustics
  - Balanced sound
  - Greater envelopment
- Open space in upper walls
- Sound arrives at ears where they are most sensitive



REVERBERATION IN A SHOEBOX HALL

Through geometric relationships, the following were determined:

$$r_1 = d_1\sqrt{\left(\frac{y}{2d_1 - x}\right)^2 + 1}$$

$$r_2 = \sqrt{\left[y\left(\frac{2d_1 - x}{d_1}\right)\right]^2 + (d_1 - x)^2}$$

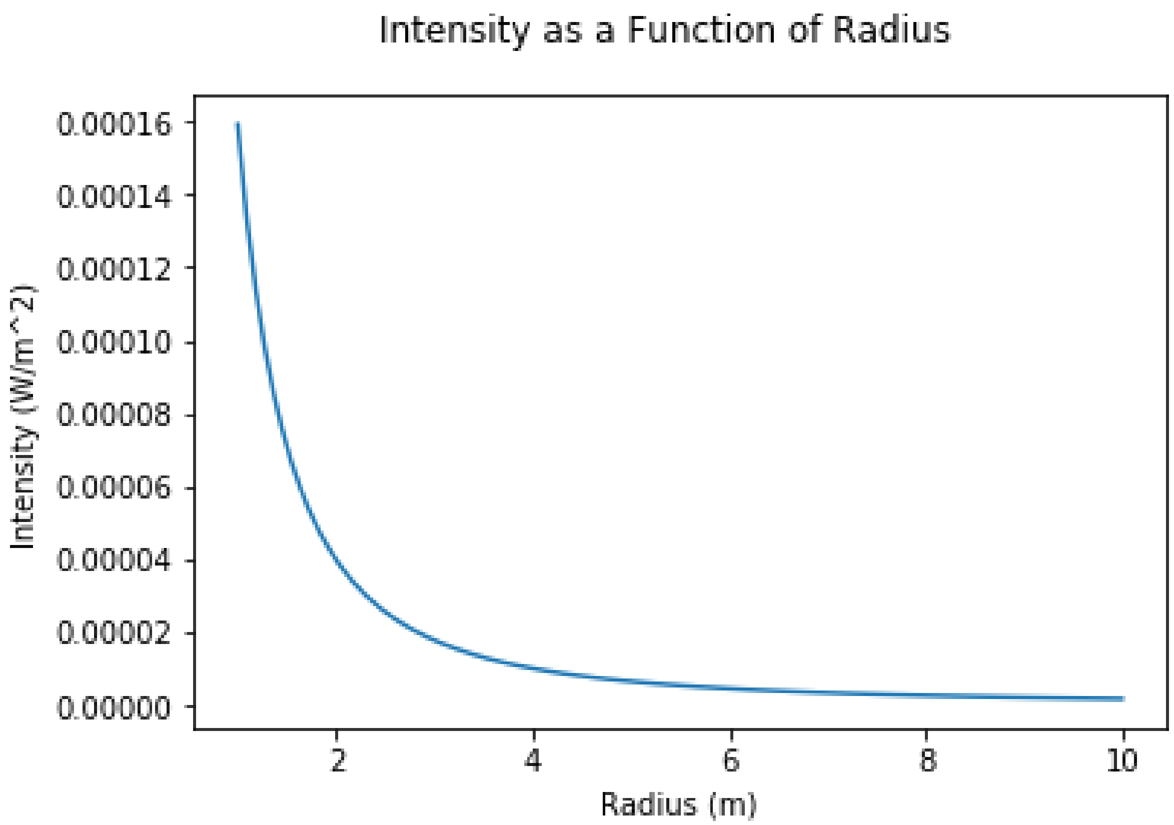
$$r_3 = d_1\sqrt{\left(\frac{y}{2d_1 + x}\right)^2 + 1}$$

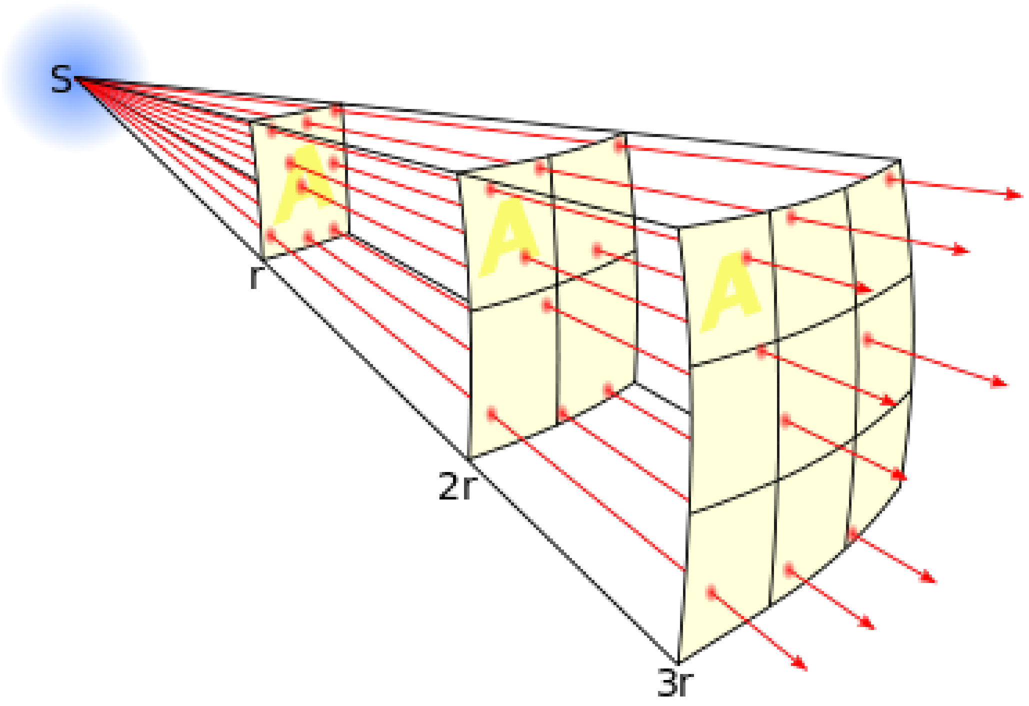
$$r_4 = \sqrt{\left[y\left(\frac{2d_1 + x}{d_1}\right)\right]^2 + (d_1 + x)^2}$$

$$r_5 = \sqrt{x^2 + y^2}$$

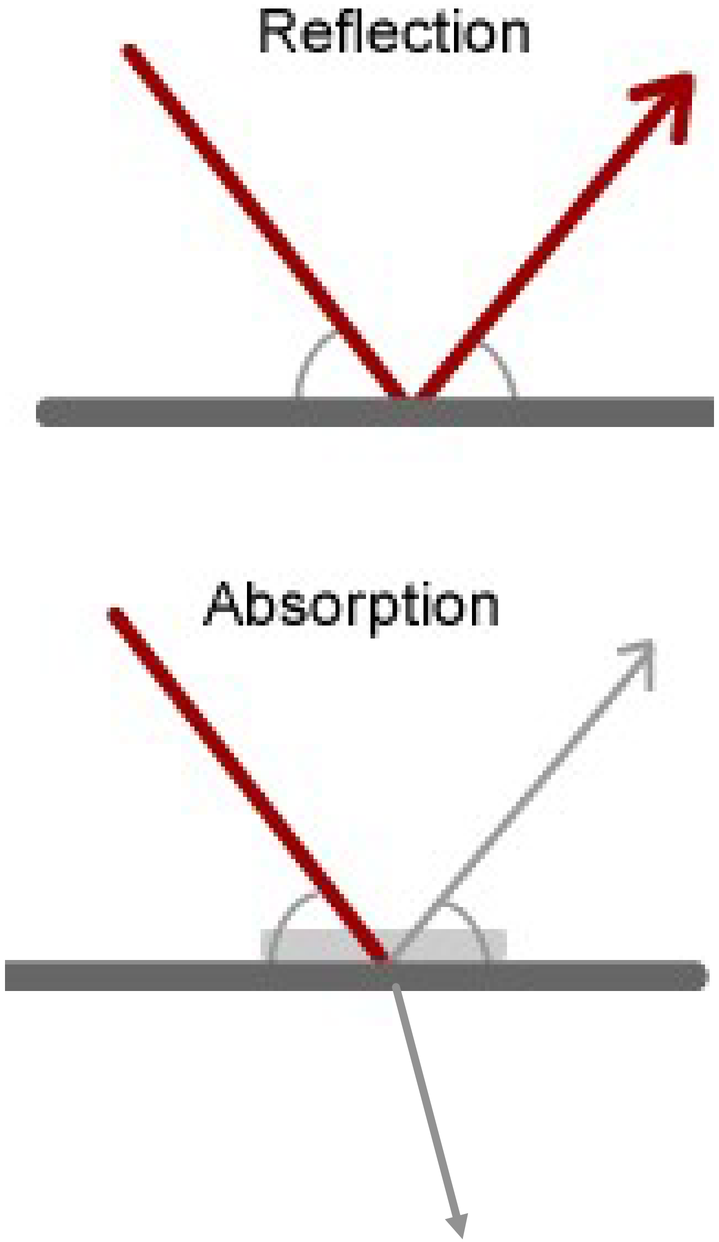
INVERSE SQUARE LAW

is modeled by the equation $I = \frac{P}{4\pi r^2}$





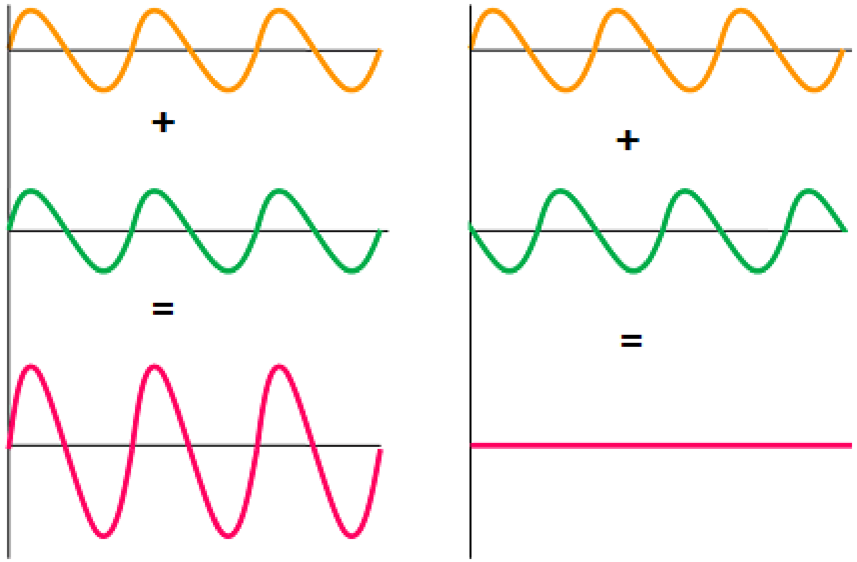
SOUND ABSORPTION AND INTERFERENCE

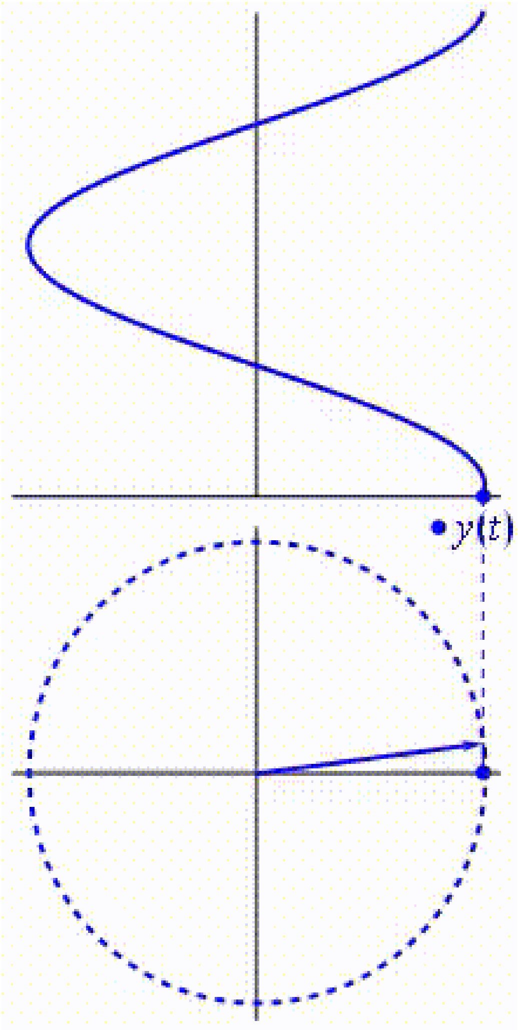


- Sound absorption occurs when a sound wave hits an absorbent material
- The sound absorption coefficient is represented as

$$\alpha = \frac{I_\alpha}{I}$$

- Destructive and constructive interference decrease or increase amplitude





PHASORS

- Phasors represent the amplitude of a certain wave at a point in time
- The amplitude of a specific path is found through the use of

$$I = \frac{1}{2}\rho V \omega^2 A^2 = \frac{P}{4\pi r^2}$$

- Solving for amplitude, the equation is rewritten as

$$A = \frac{1}{r}\sqrt{\frac{P}{2\pi\rho v\omega^2}} \text{ or } A_\alpha = \frac{1}{r}\sqrt{\left(\frac{\alpha P}{2\pi\rho v\omega^2}\right)}$$

- When split up into it’s x and y components,

$$A_x = Acos(\omega t + \phi)$$

$$A_y = Asin(\omega t + \phi)$$

DERIVING POWER

- The intensity of a sound wave, I, is modeled by
$$I = \frac{P_s}{4\pi r^2} \;,$$
- P_s is found by using the equation for sound level, where
$$\beta = 10\log\left(\frac{I}{I_0}\right)$$
- Rearranging for intensity and equating it to the inverse square law, it is found that power can be rewritten as
$$P_s = 4\pi r_c^2 I_0 10^{\beta/10}$$

MATRIX OF INTENSITIES

Plugging the total amplitude back into

$$I = \frac{1}{2}\rho V \omega^2 A_{tot}^2$$

the final intensity for a single point is found. The intensities that are calculated are then organized in a matrix

$$M = \begin{bmatrix} I_{x_1,y_m} & \cdots & I_{x_m y_n} \\ \vdots & \ddots & \vdots \\ I_{x_1 y_1} & \cdots & I_{x_m y_1} \end{bmatrix}$$

Putting Everything Into Code

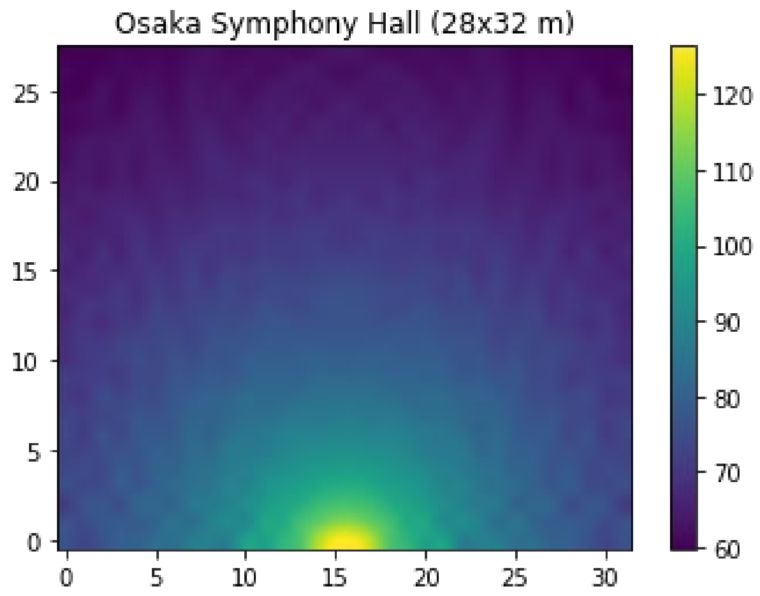
- Imported numpy, math, matplotlib

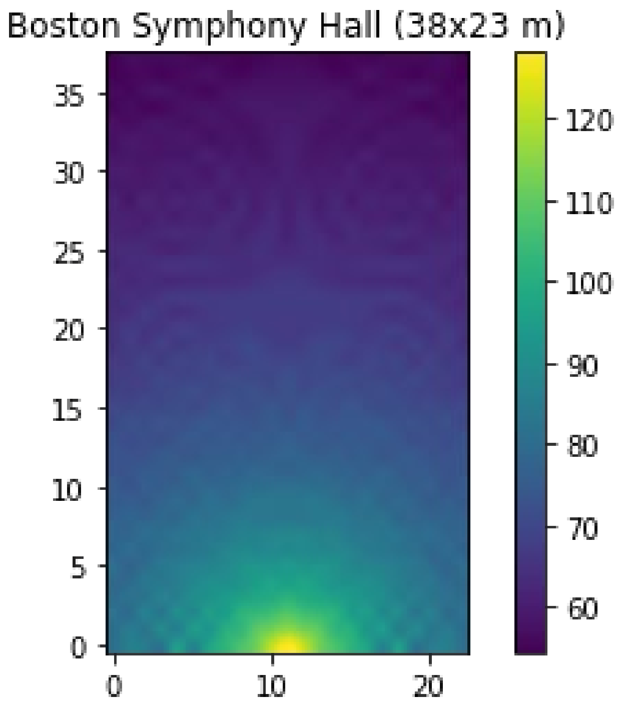
```
columns = 10
rows = 15
M = np.zeros((rows, columns))
```

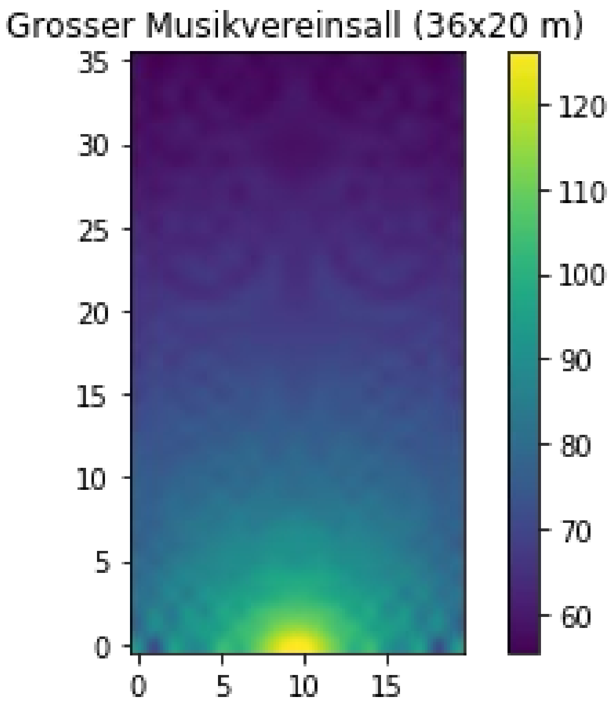
```
d = columns/2 + 0.5
x = -d
y = 0
v = 344
f = 440
P = 4 * np.pi * R**2 * 10**-12 * 10**(B/10)
p = 1.225
w = 2 * np.pi * f
Q = math.sqrt(2 * P / (p * v * w**2))
t = 0
```

```
for i in M:
    y += 1
    for j in range(len(i)):
        x += 1
        #radii
        rDirect = math.sqrt(x**2 + y**2)
        r_1 = d * math.sqrt((y / (2 * d - x))**2 + 1)
        r_2 = math.sqrt((y * (2 * d - x / d))**2 + (d - x)**2)
        r_3 = d * math.sqrt((y / (2 * d + x))**2 + 1)
        r_4 = math.sqrt((y * (2 * d + x / d))**2 + (d + x)**2)
        #path difference
        pathDif_1 = r_1 + r_2 - rDirect
        pathDif_2 = r_3 + r_4 - rDirect
        #phase difference
        phaseDif_1 = 2 * np.pi * pathDif_1 * f / v
        phaseDif_2 = 2 * np.pi * pathDif_2 * f / v
        #amplitudes
        amp_1 = Q * 1 / (r_1 + r_2)
        amp_1x = amp_1 * np.cos(w * t + phaseDif_1)
        amp_1y = amp_1 * np.sin(w * t + phaseDif_1)
        amp_1a = Q_a * 1 / (r_1 + r_2)
        amp_1ax = amp_1a * np.cos(w * t + phaseDif_1)
        amp_1ay = amp_1a * np.sin(w * t + phaseDif_1)
        A1x = amp_1x - amp_1ax
        A1y = amp_1y - amp_1ay
        amp_2 = Q * 1 / (r_3 + r_4)
        amp_2x = amp_2 * np.cos(w * t + phaseDif_2)
        amp_2y = amp_2 * np.sin(w * t + phaseDif_2)
        amp_2a = Q_a * 1 / (r_1 + r_2)
        amp_2ax = amp_2a * np.cos(w * t + phaseDif_2)
        amp_2ay = amp_2a * np.sin(w * t + phaseDif_2)
        A2x = amp_2x - amp_2ax
        A2y = amp_2y - amp_2ay
        ampDirect = Q * 1 / rDirect
        ampDirect_x = ampDirect * np.cos(w * t)
        ampDirect_y = ampDirect * np.sin(w * t)
        ampTot = (A1x + A2x + ampDirect_x)**2 + (A1y + A2y + ampDirect_y)**2
        #appending intensity
        intensity = 0.5 * p * v * w**2 * ampTot
        i[j] = 10 * math.log(intensity / 10**-12)
    x = -d
```

```
norm = ImageNormalize(M, interval = MinMaxInterval())
fig, ax = plt.subplots()
im = ax.imshow(M, interpolation = 'spline16', vmin = np.amin(M), vmax = np.amax(M), origin = 'lower')
fig.colorbar(im)
plt.title('Boston Symphony Hall')
plt.show()
```

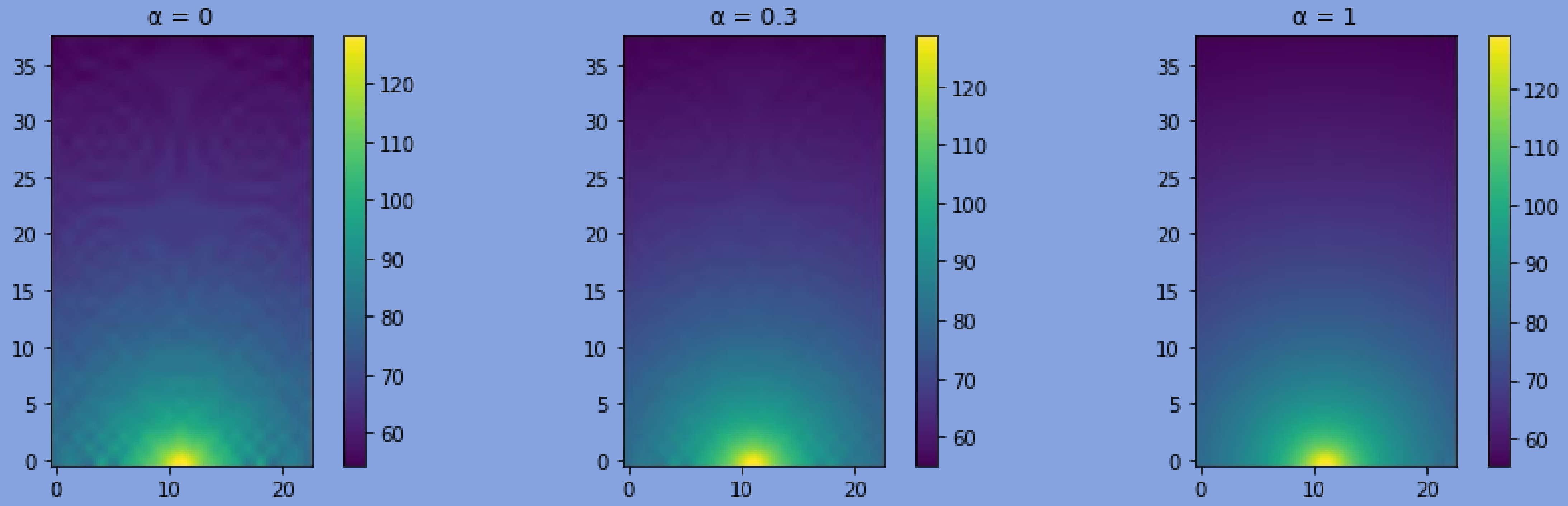






HEAT MAPS OF DIFFERENT SIZED SHOEBOX HALLS

$(\alpha = 0)$



VARYING SOUND ABSORPTION IN BOSTON HALL

REFERENCES

Long, Marshall. "What Is So Special about Shoebox Halls? Envelopment, Envelopment, Envelopment." *Acoustics Today*, 31 July 2009, acousticstoday.org/special-shoebox-halls-envelopment-envelopment-envelopment/

Barron, Mike. (1996). Loudness in concert halls. Acustica. 82. S21-S29.

Engineering ToolBox, (2003). *Room Sound Absorption - Sound Absorption Coefficient*.

Jeon, Jin Yong & Jo, Hyun-in & Kwak, Kee & Kim, Hyun. (2018). The design goal of acoustic coefficients in concert halls related to reverberation and loudness.

Hidaka, Takayuki & Nishihara, Noriko. (2015). Acoustical quality in concert halls as related to hall shape: Shoebox, surround, and other.. Psychomusicology: Music, Mind, and Brain. 25. 240-252. http://dx.doi.org/10.1037/pmu0000050240