Biomimicry research and design report

Introduction
Every year, around 20%-40% of crops in the US are lost to pests.[1] Many companies, farmers, and researchers have invested money and time into repelling these animals and have divided these techniques into 4 categories:[2]

1) Exclusion (using physical barricades to block access to crops)
2) Repel (dispersing taste or smell repellant along crops to dissuade consumption)
3) Deter (propping visual or auditory device to scare pests away from crops)
4) Attract (inviting predatory animals to hunt pests)

A study written by Charles Wilson and Gordon McKillop focusing on the Deter technique, found that using sound to scare European Rabbits only worked in close proximity to the animals and was not sustainable over large agricultural farmland.[3] In addition, a paper by Mary Bomford and Peter O'Brien found that pests quickly habituate to loud volumes, white noise, and prerecorded animal noises.[4] The purpose of this paper is to break down how bull snakes (*Pituophis catinefer sayi*) propagate and manipulate noise, so that their vocal audio signals can be used as a means of warding off prey (e.g. mice, gophers, and birds)[5] without falling into the same traps that other papers outline for pest deterrents. Specifically, we will be focusing on (1) recreating basic vocal system, (2) adding variation by finding ways to manipulate characteristics of sound, and (3) find wave of increasing propagation distance.

Biomimicry

Sound Production in Pituophis melanoleucus (Serpentes:Colubridae) With the First Description of a Vocal Cord in Snakes by Bruce Young, Stan Shift, and William Yost breaks down the bull snake's vocal system into a set of components that each influence the hissing the bellowing noise of the snake.[6] The researchers isolated these organ functions by surgically removing them and measuring how it affects the snakes' vocal noises. These components can be broken down as such: (from closest to lung to closest to mouth)

1) Tracheal Lumen ("Windpipe") – Connects to lungs and larynx
2) Laryngeal Septum ("Vocal Chord") – Non-elastic single slit structure that produces harmonic elements
3) Glottal Rim ("Aperture") – Small gap which diffracts certain frequencies of sound
4) Ricoid Cartilage – Elastic tissue that influences length and tension of Laryngeal Septum
5) Epiglottal Keel – Concave flow divider; increases amplitude of sound wave

The snake uses these components to produce 2 unique noises:

[1] (Gula, 2024)
[2] (LaLiberte, 2024)
[3] (Charles J. Wilson, 1986)
[4] (Mary Bomford, 1990)
[5] (Cosley Zoo, n.d.)
[6] (BRUCEA.YOUNG, 1995)

Hissing

- a frequency range of approximately 2,000-9,500 Hz, with lower-amplitude sounds extending down to 500 Hz
- no frequency variation or temporal modulation
- amplitudes ranging from approximately 60 to 80 dB

Bellow

- high-amplitude sound with a frequency range of 2,000-9,500 Hz, with lower-frequency sound extending down to 500 Hz, and a dominant frequency of approximately 2,600 Hz
- Initial pulse leads into a longer-duration, lower-amplitude, constant sound characterized by an acoustic range of roughly 500 to 10,000 **Hz**
- amplitudes ranging from approximately 40 to 60 dB
- hypothesize that bellowing results from an imbalance between the exhalent air pressure and the tension of the vocal cord

Solution

In order to understand how we can manipulate these elements, let us break down the movement of a sound wave and the change in its components into a simplified grid algorithm.

Components of sound wave

The volume of a sound wave is determined by its output delta density, Δd, which can be calculated as such:[7]

$$\Delta d = a * sin(2\pi T f)$$

The time elapsed, T, recognizes the time passed for some specific point of in wave which we will rewrite as $T = t + \Delta T$. In other words, time elapsed equals net time since start of algorithm plus time displacement of the point along the soundwave since emission. The frequency, f, of a sound wave which informs pitch is constant, so we can include as one of first variable input, and the amplitude can be calculated as part of another equation:

$$I = 2 * \pi^2 * \rho * f^2 * v * a^2$$

This equation uses medium density, ρ, which we will assume is equal to standard air density, $1.293\frac{kg}{m^3}$, and velocity, v, which we will assume is $343\frac{m}{s}$. The initial intensity, I_0, is our second variable input which will change during the sound waves' movement.

Gradient of sound intensity

The intensity of a sound wave can be calculated based on distance using the Inverse-Cube Law: $I_r = \frac{I_0}{4\pi r^2}$ such that the intensity is dispersed across a spherical surface based on distance.[8]

Let us now represent the intensity of sound in space as the 2-D vector, $\vec{I}_{x,y} = \begin{bmatrix} a \\ b \end{bmatrix}$, such that *x* and *y* are integers represent the x and y coordinates respectively; *a* and *b* are real numbers

[7] (Elert, 1998)
[8] (University of Central Florida, 2016)

representing the *x* and *y* components of the intensity vector. $\hat{I}_{x,y}$ (^ represents normalization) indicates the direction it is moving and $\|\vec{I}_{x,y}\|$ indicates the magnitude of intensity.

The model's algorithm will calculate a new $\vec{I}_{x,y}$ value for each slot across the entire grid per simulation step. Because the speed of intensity is constant, we only need to consider the 8 adjacent tiles for each new step. To preserve the net intensity, the intensity's influence should be weighted by direction towards each pixel. This percent direction can be calculated as the dot product between the normalized intensity vector and the unit vector pointing towards the calculated location:

E.g. $\hat{I}_{x,y} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ represents a scalar value from -1 to 1 of the intensity's percent movement in the positive-x direction

This value should then be clamped to a minimum of 0 such that an intensity moving in an opposite direction does not have negative weight and is ignored:

E.g. $max\left\{0, \hat{I}_{x,y} \circ \begin{bmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\}$ represents the intensity's clamped percent movement from 0 to 1 in the negative-x and positive-y direction.

Using this formula across all adjacent slots, here is a function, *InfluenceKernel(),* for some 3x3 2-D array calculating the weighted direction towards the center slot:

$$InfluenceKernel\left(\begin{bmatrix} \vec{I}_{x-1,y+1} & \vec{I}_{x,y+1} & \vec{I}_{x+1,y+1} \\ \vec{I}_{x-1,y} & \vec{I}_{x,y} & \vec{I}_{x+1,y} \\ \vec{I}_{x-1,y-1} & \vec{I}_{x,y-1} & \vec{I}_{x+1,y-1} \end{bmatrix}\right) = \begin{bmatrix} max\left\{0, \hat{I}_{x-1,y+1} \circ \begin{bmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y+1} \circ \begin{bmatrix} 0 \\ -1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y+1} \circ \begin{bmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} \\ max\left\{0, \hat{I}_{x-1,y} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y} \circ \begin{bmatrix} 0 \\ 0 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y} \circ \begin{bmatrix} -1 \\ 0 \end{bmatrix}\right\} \\ max\left\{0, \hat{I}_{x-1,y-1} \circ \begin{bmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y-1} \circ \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y-1} \circ \begin{bmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} \end{bmatrix}$$

The central value of the kernel always computes to 0, so it will be substituted as such for simplicity. To compute the final Intensity vector, we then convolute the original influence matrix (as passed into the function above) by the Influence Kernel.[9] For future reference, $\circledast$ represents convolution whereas $*$ represents matrix multiplication (asterisks are typically used to represent both, so it has been differed for legibility). The updated intensity function can be written in full as such:

$$\vec{I}_{x,y} = \begin{bmatrix} \vec{I}_{x-1,y+1} & \vec{I}_{x,y+1} & \vec{I}_{x+1,y+1} \\ \vec{I}_{x-1,y} & 0 & \vec{I}_{x+1,y} \\ \vec{I}_{x-1,y-1} & \vec{I}_{x,y-1} & \vec{I}_{x+1,y-1} \end{bmatrix} \circledast \begin{bmatrix} max\left\{0, \hat{I}_{x-1,y+1} \circ \begin{bmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y+1} \circ \begin{bmatrix} 0 \\ -1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y+1} \circ \begin{bmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} \\ max\left\{0, \hat{I}_{x-1,y} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\} & 0 & max\left\{0, \hat{I}_{x+1,y} \circ \begin{bmatrix} -1 \\ 0 \end{bmatrix}\right\} \\ max\left\{0, \hat{I}_{x-1,y-1} \circ \begin{bmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y-1} \circ \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y-1} \circ \begin{bmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} \end{bmatrix}$$

Time displacement

Going back to the original sound equation, we also need to calculate the time displacement, $\Delta T_{x,y}$, for each slot of the grid. This can also be thought of as a third intensity vector component, but it should be ignored for aforementioned formulae. Using our $\vec{I}_{x,y}$ vector we can isolate the direction of greatest influence by subtracting the distances between the calculated

[9] (Shrestha, 2023)

intensity and slot directions and choosing the corresponding direction for whichever distance is smallest:

$$greatestDirectionInfluence(\vec{I}_{x,y}) = min\left\{\left\|\vec{I}_{x,y} - \begin{bmatrix}\sqrt{0.5}\\ -\sqrt{0.5}\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}0\\ -1\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}-\sqrt{0.5}\\ -\sqrt{0.5}\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}1\\ 0\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}-1\\ 0\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}\sqrt{0.5}\\ \sqrt{0.5}\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}0\\ 1\end{bmatrix}\right\|, \left\|\vec{I}_{x,y} - \begin{bmatrix}-\sqrt{0.5}\\ \sqrt{0.5}\end{bmatrix}\right\|\right\}$$

The new $\Delta T_{x,y}$value is set to the $\Delta T_{x,y}$ value from the direction of greatest influence plus the delta time. Delta time is a constant representing the time it takes from intensity to move between 2 adjacent slots based on the fixed velocity: Δt, for lateral movements and $\Delta t * \sqrt{2}$ for diagonal movements to account for the extra distance.

Lateral time displacement: $\Delta T_{x,y} = \Delta T_{x\pm1,y\pm1} + \Delta t$

Diagonal time displacement: $\Delta T_{x,y} = \Delta T_{x\pm1,y\pm1} + \left(\Delta t * \sqrt{2}\right)$

Sound colliders

When touching a collider, the sound intensity should reflect opposite to the direction of the collider while losing a portion of its intensity based on its elasticity. The elasticity, ε, represents a percentage (from 0 to 1) of the intensity maintained during reflection. Similarly to the sound intensity, colliders are held in slots and only have normal surfaces for the 8 adjacent directions (rendered as squares but function as octagons). Within these functions x and y represent components instead of grid positions, and the sign of $\pm\varepsilon$ should be opposite to its corresponding component factor.

Horizontal Collisions: $\vec{I} = \begin{bmatrix}\left|\vec{I}_x\right| * (\pm\varepsilon)\\ \vec{I}_y * \varepsilon\\ \Delta T\end{bmatrix}$

Vertical Collisions: $\vec{I} = \begin{bmatrix}\vec{I}_x * \varepsilon\\ \left|\vec{I}_y\right| * (\pm\varepsilon)\\ \Delta T\end{bmatrix}$

Diagonal Collisions: $\vec{I} = \begin{bmatrix}\left|\vec{I}_x\right| * (\pm\varepsilon)\\ \left|\vec{I}_y\right| * (\pm\varepsilon)\\ \Delta T\end{bmatrix}$

Non-destructive sound intensity interference

The current issue with this approach to sound intensity movement is that the sound wave loses intensity when colliding with other waves. While sound waves do have destructive interference (we will return to this later), this destruction only affects the net amplitude output, whereas the waves' intensities should be maintained and pass through each other undampened.[10] To account for this, let us add a fourth dimension to our intensity vector and assume that multiple intensities of differing interference layers can exist in the same slot of the grid. Our new intensity vector can be broken down as such:

$$\vec{I}_{x-location,y-location} = \begin{bmatrix}x - component\ of\ intesity\\ y - component\ of\ intensity\\ time\ displacement\\ interference\ layer\end{bmatrix}$$

[10] (Byrne, 2013)

For any formulae using $\vec{I}_{x,y}$ let us assume that only the x and y components are operated on unless otherwise specified. Because our system only accounts for 8 directions of movement, we can limit the interference layers to a set of integers between 0 and 7 inclusive. This also limits us to a maximum of 8 intensity vectors for each grid slot. We will also need to adjust previous formulae to account for this layering: (only the x and y components of intensity are operated upon)

$$\text{Layer 0: } \vec{I}_{x,y} = \begin{bmatrix} \vec{I}_{x-1,y+1} & \vec{I}_{x,y+1} & 0 \\ \vec{I}_{x-1,y} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \circledast \begin{bmatrix} max\left\{0, \hat{I}_{x-1,y+1} \circ \begin{bmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y+1} \circ \begin{bmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} & 0 \\ max\left\{0, \hat{I}_{x-1,y} \circ \begin{bmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\text{Layer 1: } \vec{I}_{x,y} = \begin{bmatrix} \vec{I}_{x-1,y+1} & \vec{I}_{x,y+1} & \vec{I}_{x+1,y+1} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \circledast \begin{bmatrix} max\left\{0, \hat{I}_{x-1,y+1} \circ \begin{bmatrix} 0 \\ -1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y+1} \circ \begin{bmatrix} 0 \\ -1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y+1} \circ \begin{bmatrix} 0 \\ -1 \end{bmatrix}\right\} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\text{Layer 2: } \vec{I}_{x,y} = \begin{bmatrix} 0 & \vec{I}_{x,y+1} & \vec{I}_{x+1,y+1} \\ 0 & 0 & \vec{I}_{x+1,y} \\ 0 & 0 & 0 \end{bmatrix} \circledast \begin{bmatrix} 0 & max\left\{0, \hat{I}_{x,y+1} \circ \begin{bmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y+1} \circ \begin{bmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} \\ 0 & 0 & max\left\{0, \hat{I}_{x+1,y} \circ \begin{bmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{bmatrix}\right\} \\ 0 & 0 & 0 \end{bmatrix}$$

$$\text{Layer 3: } \vec{I}_{x,y} = \begin{bmatrix} \vec{I}_{x-1,y+1} & 0 & 0 \\ \vec{I}_{x-1,y} & 0 & 0 \\ \vec{I}_{x-1,y-1} & 0 & 0 \end{bmatrix} \circledast \begin{bmatrix} max\left\{0, \hat{I}_{x-1,y+1} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\} & 0 & 0 \\ max\left\{0, \hat{I}_{x-1,y} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\} & 0 & 0 \\ max\left\{0, \hat{I}_{x-1,y-1} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\} & 0 & 0 \end{bmatrix}$$

$$\text{Layer 4: } \vec{I}_{x,y} = \begin{bmatrix} 0 & 0 & \vec{I}_{x+1,y+1} \\ 0 & 0 & \vec{I}_{x+1,y} \\ 0 & 0 & \vec{I}_{x+1,y-1} \end{bmatrix} \circledast \begin{bmatrix} 0 & 0 & max\left\{0, \hat{I}_{x+1,y+1} \circ \begin{bmatrix} -1 \\ 0 \end{bmatrix}\right\} \\ 0 & 0 & max\left\{0, \hat{I}_{x+1,y} \circ \begin{bmatrix} -1 \\ 0 \end{bmatrix}\right\} \\ 0 & 0 & max\left\{0, \hat{I}_{x+1,y-1} \circ \begin{bmatrix} -1 \\ 0 \end{bmatrix}\right\} \end{bmatrix}$$

$$\text{Layer 5: } \vec{I}_{x,y} = \begin{bmatrix} 0 & 0 & 0 \\ \vec{I}_{x-1,y} & 0 & 0 \\ \vec{I}_{x-1,y-1} & \vec{I}_{x,y-1} & 0 \end{bmatrix} \circledast \begin{bmatrix} 0 & 0 & 0 \\ max\left\{0, \hat{I}_{x-1,y} \circ \begin{bmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} & 0 & 0 \\ max\left\{0, \hat{I}_{x-1,y-1} \circ \begin{bmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y-1} \circ \begin{bmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} & 0 \end{bmatrix}$$

$$\text{Layer 6: } \vec{I}_{x,y} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \vec{I}_{x-1,y-1} & \vec{I}_{x,y-1} & \vec{I}_{x+1,y-1} \end{bmatrix} \circledast \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ max\left\{0, \hat{I}_{x-1,y-1} \circ \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x,y-1} \circ \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y-1} \circ \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\} \end{bmatrix}$$

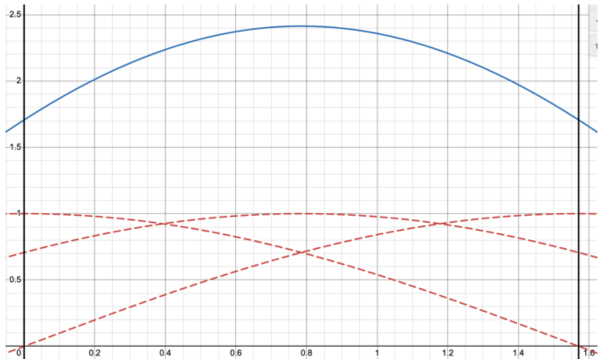
$$\text{Layer 7: } \vec{I}_{x,y} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \vec{I}_{x+1,y} \\ 0 & \vec{I}_{x,y-1} & \vec{I}_{x+1,y-1} \end{bmatrix} \circledast \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & max\left\{0, \hat{I}_{x+1,y} \circ \begin{bmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} \\ 0 & max\left\{0, \hat{I}_{x,y-1} \circ \begin{bmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} & max\left\{0, \hat{I}_{x+1,y-1} \circ \begin{bmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{bmatrix}\right\} \end{bmatrix}$$

The calculation for time displacement is the same, but only on a layer per layer basis. When accounting for sound wave colliders, the layer should be switched to the layer in which the direction is reflected across the axis of collision including time displacement.

Adjusting for gradient explosion

An issue with these formulae is that net intensity is not conserved but increases across all layers. Figure 1 shows 4 functions: the 3 dotted red lines represent the resulting 3 dot products of layer 1's influence kernel across varying direction (0 to $\frac{\pi}{2}$ radians) and the blue line representing the sum of these values. The new intensity is the product of the incoming intensity and the blue line as shown. The graph demonstrates that the sum is greater than 1 across the entire range of 0 to $\frac{\pi}{2}$ which indicates an increase in intensity and an exploding gradient across the entire grid.

Figure 1: Resulting dot products of intensity kernel and sum



To adjust for this explosion, here are 2 proposed formulae: $\left(max\left\{0, \hat{I}_{x,y} \circ \hat{d}\right\}\right)\left(\sqrt{2} - 1\right)$ as shown in figure 2 and $\left(max\left\{0, \hat{I}_{x,y} \circ \hat{d}\right\}\right)^6(0.8)$ as shown in figure 3 ($\hat{d}$ is direction towards center as used in influence kernel). Both of these expressions limit the new intensity to a maximum of 1, but figure 2 has a greater falloff than figure 3. I found figure 3 to produce better looking results, but both functions are worth consideration and potential reevaluation.

Figure 2: Sum of $\left(max\left\{0, \hat{I}_{x,y} \circ \hat{d}\right\}\right)(\sqrt{2} - 1)$

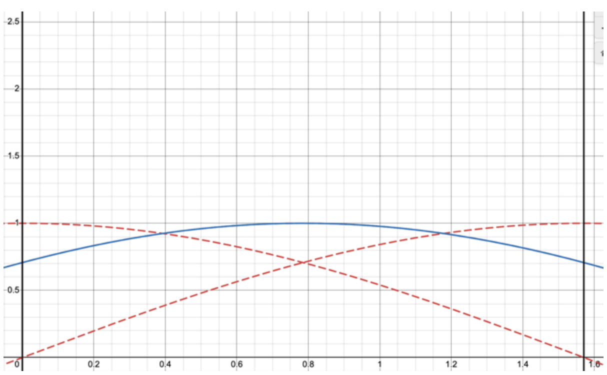
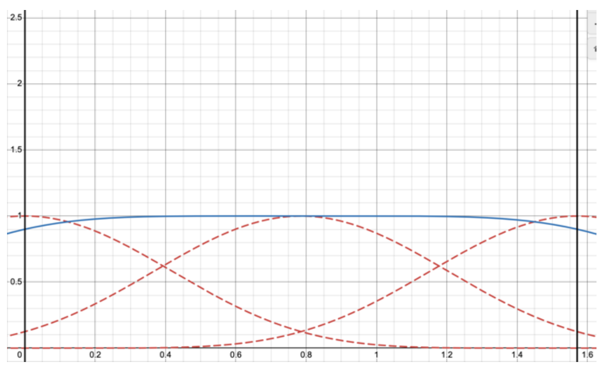


Figure 3: Sum of $\left(max\left\{0, \hat{I}_{x,y} \circ \hat{d}\right\}\right)^6(0.8)$



Simulation algorithm

The relationship between distance meters (d_m), speed of sound ($v = 343\,\frac{m}{s}$), Grid Pixel Distance ($\boldsymbol{d_u}$), and Delta time per grid pixel (Δt) can be written as such:

$$d_m = v * d_u * \Delta t$$

Using this new equation we can isolate a few input variables:

Initial Frequency – f_0
Elasticity – ε
Grid Pixel Distance – $\boldsymbol{d_u}$ (resolution of simulation)
Delta time per grid pixel (seconds per pixel) – $\Delta\boldsymbol{t}$

A build of the model can be access here Dropbox Link (I originally planed to host online, but WebGL does not support compute shaders which my model runs on)

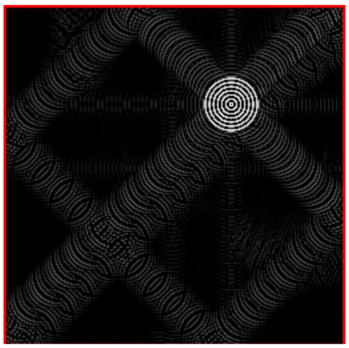


Figure 1: Example of model running without colliders (the diagonals seem to not be dispersing properly)

In an attempt to recreate the vocal organs as detailed in the biomimicry section here are a few model results.

Figure 2: Slit of Epiglottal Keel

Figure 3: Flap opening of Laryngeal Septum

Discussion

Looking at the results visibly, the structure of the snake vocal organs seem to be scalable in allowing for wider distances and can be altered for finer control over sound components.

The model is able to effectively communicate large scale changes and areas of intensity but lacks specificity in terms of measuring the exact intensity at some point in space. The model also fails to emulate radial patterns between the 8 directions but instead seems to create beams heading in those 8 directions without properly diffusing. This artifact can likely be reduced by rewriting how time displacement is calculated for these angles, changing the factor of the influence kernel for adjacent values, or some mix of the 2.

As a limitation of time for this paper the simulation has 2 large simplifications:
1) Elasticity is a global constant and cannot be assigned on a collider per collider basis. As explained in the biomimicry section, the tissue of the snake's vocal track varies greatly in elasticity in order to produce different effects, but this model only uses a single elasticity value for all collisions.
2) Frequency does not diffract but instead remains constant. In snakes as well as humans, we are able to change the pitch of our voice by manipulating the diffraction of frequency, but this is not addressed in the model.

References

Works Cited

BRUCEA.YOUNG, S. W. (1995). Sound Production in Pituophis melanoleucus (Serpentes:Colubridae) With the First Description of a Vocal Cord in Snakes. *THE JOURNAL OF EXPERIMENTALZOOLOGY*.

Byrne, C. L. (2013). *Mathematics of Signal Processing: A First Course.* Lowell: University of Massachusetts.

Charles J. Wilson, I. G. (1986). An Acoustic Scaring Device Tested against European Rabbits. *Wildlife Society Bulletin*.

Cosley Zoo. (n.d.). *Pituophis catenifer sayi*. Retrieved from Cosley Zoo: https://cosleyzoo.org/common-bullsnake/#:~:text=In%20the%20wild%2C%20bullsnakes%20primarily,bullsnake%20eats%20mice%20and%20chicken.

Elert, G. (1998). *Sound Intensity*. Retrieved from The Physics Hypertextbook: https://physics.info/intensity/

Gula, L. T. (2024, September 16). *Researchers Helping Protect Crops From Pests*. Retrieved from National Institute of Food and Agriculture: https://www.nifa.usda.gov/about-nifa/blogs/researchers-helping-protect-crops-pests

LaLiberte, K. (2024, March 11). *Keeping Animal Pests Out of Your Garden*. Retrieved from Gardeners: https://www.gardeners.com/how-to/keep-animals-out-of-your-garden/5426.html?srsltid=AfmBOorhOrsFPGjCtEEMw71WR3U_PIb5SxxrmozZ0Paf9F7OfCS0nlJx

Mary Bomford, P. H. (1990). Sonic Deterrents in Animal Damage Control: A Review of Device Tests and Effectiveness. *Wildlife Society Bulletin*.

Shrestha, A. (2023, March 23). *Understanding the Differences Between Convolution and Matrix Multiplication*. Retrieved from Medium: https://aishwaryashrestha232.medium.com/understanding-the-differences-between-convolution-and-matrix-multiplication-a-guide-to-master-2d-

9d1a2162cd30#:~:text=Convolution%20and%20Matrix%20Multiplication%20processes,is%20the%20second%20matrix%20transposed).

University of Central Florida. (2016). Energy and Power of a Wave. In *UNIVERSITY PHYSICS VOLUME 1.* Retrieved from University of Central Florida: https://pressbooks.online.ucf.edu/osuniversityphysics/chapter/16-3-wave-speed-on-a-stretched-string/