Implementing Curved Bicone in GENETIS Genetic Algorithm

Leo Deer †

Introduction

The GENETIS Collaboration created software that uses genetic algorithms to improve bicone antenna designs for ultra-high neutrino detection in Antarctica. This software starts with a base population of antenna designs, it tests the response of these antennas using finite difference time domain software, these results are used to generate a fitness score of each design which are then used to generate a new generation of antennas. This process is repeated over many generations help find an ideal antenna design. More information of the project may be found in [RMP+21]. During my time with the GENETIS collaboration, I worked to expand the complexity of possible designs generated. Originally, the software only generated symmetric bicone designs using four parameters: bicone separation, inner radius, length, opening angle. My first task was to expand the design to an asymmetric bicone which used eight parameters, the four parameters above, but on each cone of the bicone. This work was implemented in version of the software used in [RMP+21]. Now, I have expanded the software to allow for curved sides to the bicones, see Figure 1.

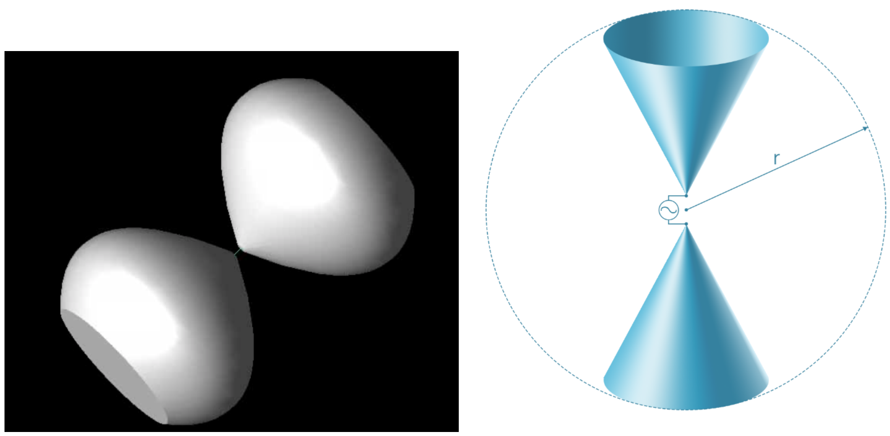


Figure 1: Left: Example of new curved bicone design. Right: Example of old bicone design.

Method

In order to properly discuss the changes implemented in the curved bicone project, it is worthwhile to take a slight detour to discuss how XFDTD[1] creates the current bicone geometry. Simply, XFDTD creates two circles (one for each bicone) with radii equal to the inner radius parameter and extrudes them normal to their surfaces see figure 2. If we consider this geometry in cylindrical coordinates (s, ϕ, z), the circular base is in the $s-\phi$ plane, while the extrusion is along the $z-$axis. This coordinate system is important for later.

† leo99deer@gmail.com

[1] XFDTD is the finite difference time domain software used by the GENETIS collaboration to create the antenna geometry and simulate the antenna response.

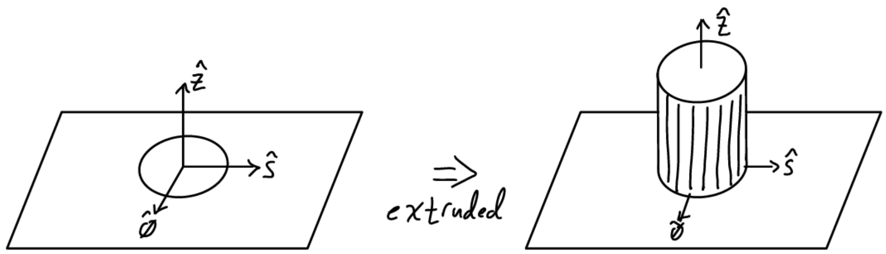


Figure 2: How XFDTD creates a Bicone

For the old bicone designs, XFDTD created bicones that were extruded at a constant angle θ. This angle is measured with respect to the $z-$axis. If we consider an $s - z$ cross section of the bicone, we get figure 3.

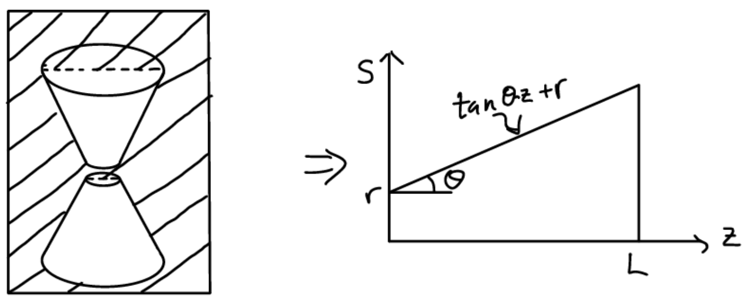
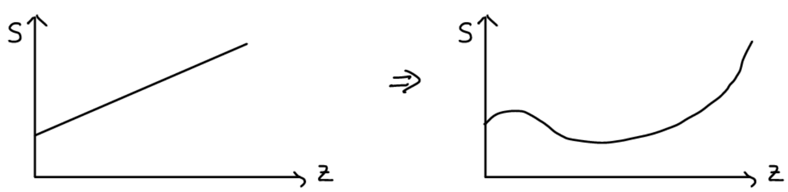


Figure 3: Cross section of old bicones

We can interpret this sketch as a linear function $s(z) = r + \tan(\theta) \cdot z$, where r is the inner radius, and $s(z)$ is the radius of our bicone at some z . To get more interesting cross sections we can think of $s(z)$ as some arbitrary function instead of just a linear one.



To genetically evolve this arbitrary function, we need to encode it in a set of parameters. To do this, we may expand this function as series. For the time being, we will drop terms of $\mathcal{O}(z^3)$ [2] i.e. $s(z) = r + \beta \cdot z + \alpha \cdot z^2$. So, now we'll evolve the α and β terms instead of θ.

Constraints

Some values of α and β cause the XFDTD simulation to terminate, so we must apply constraints on these parameters. I found that if at any point the bicone self-intersects itself, the simulation will terminate. To visually understand what self intersection means, consider the cross section in Figure 4.

[2] To improve the bicone design even more, I suggest including the $z^3, z^4, ...$ terms in the series. The implementation of these follows from what I'll show for z^2

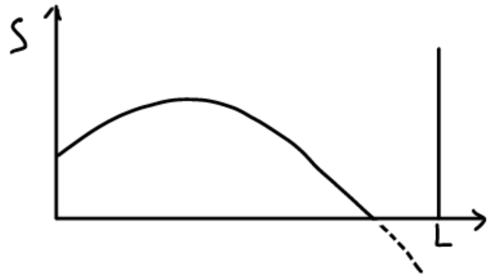
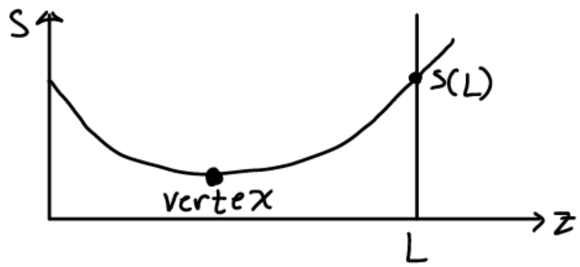


Figure 4: Cross section of self-intersecting bicone

Notice that the graph dips below the $z-$axis before $z = L$ where L is the antenna length. This is what we mean by self-intersection. The way we avoid self-intersection is demanding that the absolute minimum of our side profile $s(z)$ on the interval $z : [0, L]$ is greater than 0. From calculus we know that on a finite interval the extrema of a function occur when the first derivative is 0 or at the end points. For a parabola, the first derivative is 0 at the vertex, $(z, s) = (-\frac{\beta}{2\alpha}, r - \frac{\beta^2}{4\alpha})$

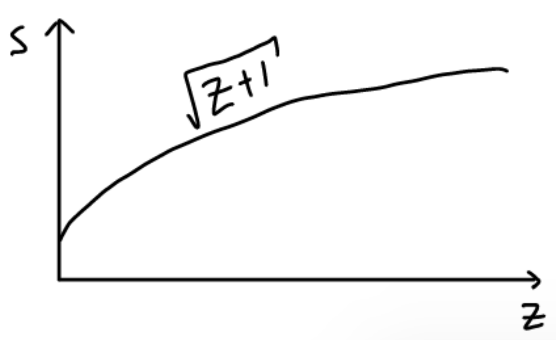


So, we have two[3] conditions which constrain α and β to avoid self-intersection.

1) if $-\frac{\beta}{2\alpha} < L$ then $r - \frac{\beta^2}{4\alpha} > 0$
2) $r + \alpha \cdot L + \beta \cdot L^2 > 0$

Implementation

The initial values of α and β we set in the genetic algorithm were based off the curve, $s(z) = \sqrt{z+1} \approx 1 + \frac{z}{2} - \frac{z^2}{8}$. So we'll use $\alpha = -\frac{1}{8}$ and $\beta = \frac{1}{2}$



[3] If we included the z^3 term then we would have three conditions. One for each vertex and the end point

In part A of the software, we need to modify the genetic algorithm itself. To do so, we replaced the θ gene[4] with the α coefficient, and we added another gene for the β coefficient. To make sure we avoid self-intersection, we created a `while` loop during the initialization for each antenna's parameters that is passed only when the conditions to avoid self-intersection are met. But, since these two conditions are not dependent on any one gene, but the relationship of all the genes, it is a condition on the chromosomes[5]. So, we implemented the `while` loop within the NPOP[6] `for` loop but outside the NSECTIONS[7] `for` loop, so that each chromosome must satisfy the condition.

In part E we edited the `fitnessFunction` to calculate the outer radius as, $R = r + \beta \cdot + \alpha \cdot L^2$

In part F of the software we changed the LRTS (length, radius, theta, separation) plot to LRAB (length, radius, α, β) which plots α and β instead of θ. β is plotted without modification. But for α, we use this to plot the θ_i the initial angle each antenna makes with the $z-$axis at $z = 0$ to help with visualization and comparison with the old bicones.

References

[RMP+21] J. Rolla, A. Machtay, A. Patton, W. Banzhaf, A. Connolly, R. Debolt, L. Deer, E. Fahimi, E. Ferstle, P. Kuzma, C. Pfendner, B. Sipe, K. Staats, and S. A. Wissel. Using evolutionary algorithms to design antennas with greater sensitivity to ultra high energy neutrinos. 2021.

[4]Gene is a term used in genetic algorithms to denote a single parameter.

[5]Like in actual genetics, a chromosome holds the information of many genes. In our case, all the parameters on one side of the bicone antenna

[6]NPOP is the number of antennas populated each generation.

[7]NSECTIONS is the number of chromosomes in each antenna. The bicone has two.