

Echo-location

1^{ro} Diego Fung

Ingeniería en computación

Instituto Tecnológico de Costa Rica

San José, Costa Rica

2019308467@estudiantec.cr

2^{do} Johel Pérez

Ingeniería en computación

Instituto Tecnológico de Costa Rica

San José, Costa Rica

pfjohelu@estudiantec.cr

Abstract—El siguiente documento es la documentación del segundo proyecto de programación correspondiente a análisis de algoritmos el cual dicta que tenemos que crear un sonar con ecolocalización utilizando Monte Carlo.

Index Terms—Monte Carlo, Algoritmo, Sonar, Ecolocalización, Python, Sonograma, Ultrasonido

I. INTRODUCCIÓN

Al iniciar a leer el problema notamos que teníamos 4 bloques de problemas grandes por resolver:

- La interfaz gráfica
- El lanzamiento de rayos
- El rebote en los muros y rayos secundarios
- La implementación de Monte Carlo

Teniendo esto en cuenta se empezó a trabajar, se creó una escena básica y se empezó a experimentar con los aspectos de detección de muros.

También se contaba con hacer un trabajo relacionado el cual consta de hacer una investigación sobre la importancia de la ecolocalización en la tecnología a continuación se explorara esa sección.

II. TRABAJO RELACIONADO

Para entender los usos de la ecolocalización primero se necesita saber que es la ecolocalización según la Real

Academia Española es la "Localización de un objeto mediante la reflexión de ondas sonoras, utilizada por especies animales como los murciélagos y los cetáceos y en los sistemas de sonar."

Al tener definida la ecolocalización vemos que es primero que todo utilizada por especies animales, esta es utilizada para que los animales puedan conocer su entorno por medio de sonidos, esto es útil para los humanos de las siguientes maneras:

1) *Sistemas de sonar*: Sonar significa: "Sound Navigation and Ranging", este es principalmente utilizado para detectar objetos los cuales se encuentran debajo del agua, esto es usado por barcos como se puede ver en la siguiente imagen:

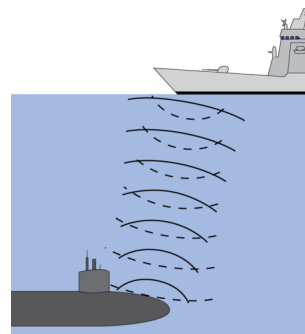


Fig. 1. Sonar detectando un submarino

2) *Sonogramas/Ultrasonidos*: Los sonogramas o ultrasonidos son otra manera con la cual la ecolocalización ayuda

a la tecnología, este es usado para crear imágenes de dentro del cuerpo humano usando el sonido para crear una imagen visible una de las cuales se puede ver en la siguiente imagen:



Fig. 2. Un ultrasonido

3) *Cálculo de puntos dado un ángulo:* Además de entender porque la ecolocación es importante debíamos investigar sobre el cálculo de puntos, ya que esta es una fórmula puramente matemática, llegamos a esta pregunta en StackExchange la cual se titula "Calculate point, given x, y, angle, and distance" y la respuesta dada por David Mitra la cual es " $x=5\cos z$, and $y=5\sin z$ " donde z es el ángulo tenido, esto nos permite calcular el ángulo de rebote

De esta manera podemos observar dos maneras por las cuales la ecolocalización es importante para la tecnología, además de ver que utilizamos para los ángulos y con esta comprensión podemos proceder con el problema principal.

III. MÉTODOS

En la sección de métodos tenemos un desglose por problema de como se intentó resolver cada uno y la lógica que hay detrás de las soluciones más que solo dar el código fuente.

A. La interfaz gráfica

Para la interfaz gráfica se decidió utilizar Python con Qt(PyQt5) debido a su versatilidad, al iniciar se creó una ventana de 500x500 y se puso solo un tipo de muros para tener un demo de las colisiones, además el sonar el cual puede ser visto a continuación:

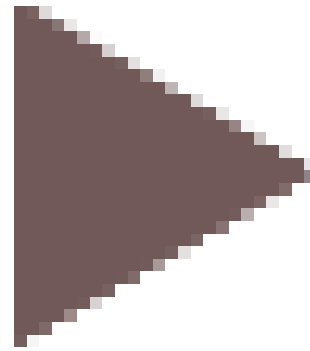


Fig. 3. Sonar

Al tener un framework con las colisiones se decidió agregar dos tipos más de muro: Durante la ejecución estas figuras son



Fig. 4. Tipo uno de muro

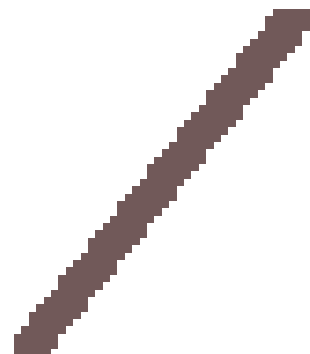


Fig. 5. Tipo dos de muro



Fig. 6. Tipo tres de muro

ocultas menos el sonar sin embargo están ahí y esto es lo que crea los diferentes tipos de rebote.

B. El lanzamiento de rayos

Cuando hablamos de este problema se puede dividir en dos problemas.

1) *Rayo principal*: Al crear el rayo principal se tenían que tomar en cuenta dos cosas:

- El ángulo de lanzamiento
- La intensidad con la que el rayo choca

El ángulo fue resuelto con una variable la cual guarda en que grado se encuentra el sonar en este momento, luego de esto se revisa cuál es el ángulo al lanzar un rayo con diferentes verificaciones, si este es 0,90,180,270 o si el ángulo se encuentra entre 0 y 90, 90 y 180, 180 y 270 o 270 y 355, es importante revisar contra 355, ya que cuando la variable llega a 360 esta se devuelve a 0.

La intensidad fue resuelta debido al movimiento del rayo, a la intensidad se le da un valor entero el cual va decreciendo dependiendo de en cuanto se vaya moviendo, el rayo se mueve píxel por píxel hasta toparse con un muro o quedarse sin energía.

2) *Colisiones*: Las colisiones se resolvieron primero buscando en la escena por los muros, al tener las coordenadas de los muros se guardan en una variable para ser utilizadas luego, cuando el rayo se está moviendo se compara la coordenada actual con las coordenadas de muro para detectar una colisión, esto también se compara con los límites de la escena para ver si el rayo choca con el final de la escena.

C. El rebote en los muros y rayos secundarios

Para el rebote en los muros utilizamos la fórmula dada anteriormente $x = 5 \cos(z)$ y $y = 5 \sin(z)$ este cálculo lo hacemos dado el número de rayos secundarios, esto es para calcular el ángulo de rebote de cada rayo secundario cuando un rayo choca con un muro, de esta manera garantizamos que

cada rayo tenga su propio ángulo, a los rayos secundarios se les comparte las colisiones de un rayo principal

D. La implementación de Monte Carlo

Empezando con la definición de Monte Carlo, Monte Carlo es un algoritmo el cual usa datos a los azares para intentar encontrar una aproximación aceptable al problema buscado, la manera de hacer esto con un sonar es dado la cantidad de rayos, es escoge una cantidad de rayos al azar para saber cuál sería el valor aproximado de rayos que se necesitan utilizar para menor costo y una pintura del entorno eficiente. Entonces Monte Carlo es aplicado a la cantidad de rayos secundarios producidos para determinar este valor.

IV. ANÁLISIS DE RESULTADOS

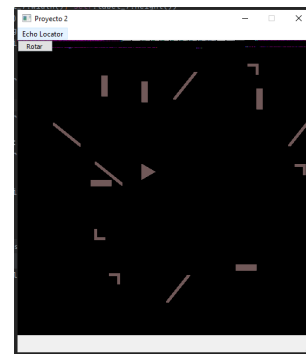


Fig. 7. Solo muros



Fig. 8. Solo pixeles

Podemos ver que el programa funciona efectivamente y podemos hacerle breakdown por todos los funcionamientos.



Fig. 9. Solo rayos principales

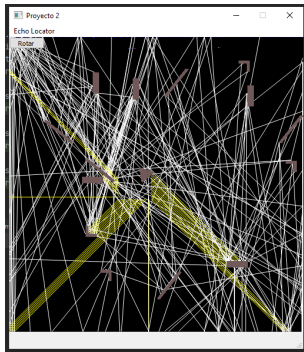


Fig. 10. Todos los rayos

Si vemos la figura 7 se puede ver la creación de la escena, esto no muestra el funcionamiento, pero será importante para más adelante.

Si vemos la figura 8 se puede ver que efectivamente localiza los muros y los reconoce con píxeles ya con estos escondidos.

Si vemos la figura 9 se puede ver como funcionan los rayos iniciales que hacen esto posible y con cuáles muros chocan directamente.

Si vemos la figura 10 todo se hace más claro, esto explica los rebotes y como píxel se pone donde debería estar.

Entonces podemos concluir que los sistemas principales si funcionan en una corrida como deberían y hacen lo requerido y con el algoritmo de montecarlo se puede determinar cuantos rayos secundarios se aproximan a los optimos, en el caso de una energía de 500 los rayos se acercan a 5.

V. CONCLUSION

Para concluir nos gustaría dar un patrón de como resolver este tipo de problema

- Lo primero es crear una escena base para hacer testing
- Lo segundo es programar un sistema de colisión para poder detectar muros
- Lo tercero seria programar el rebote dada la fórmula de puntos
- Lo ultimo sería la implementación de Monte Carlo

La solución es preferible sea en este orden para tener todo funcional antes de implementar Monte Carlo

REFERENCES

- [1] Yuen Law (2020), *Algoritmos Probabilísticos – Semana 8*.
- [2] ¿Qué es un Sonograma/Ultrasonido? - Pre-Vue.
[http : //pre – vue.com/que – es – un – sonogramaultrasonido/](http://pre-vue.com/que-es-un-sonogramaultrasonido/)
- [3] Cómo funciona el Sonar.
[https://deepersonar.com/es/es_es/como – funciona/como – funciona – el – sonar](https://deepersonar.com/es/es_es/como-funciona/como-funciona-el-sonar)
- [4] Foundation, C. Ultrasound.
<https://www.ck12.org/c/physics/ultrasound/lesson/Ultrasound-MS-PS/>
- [5] How Do Ultrasounds Work?
<https://www.ultrasound-schools.info.com/ultrasound-technology/>
- [6] López, J. Simulación de Montecarlo - Definición, qué es y concepto — Economipedia
<https://economipedia.com/definiciones/simulacion-de-montecarlo.html>
- [7] ASALE, R. ecolocalización — Diccionario de la lengua española.
<https://dle.rae.es/ecolocalizaci%C3%B3n?m=form>
- [8] Mitra, D. (2012). Calculate point, given x, y, angle, and distance
<https://math.stackexchange.com/questions/143932/calculate-point-given-x-y-angle-and-distance>