

ALGORITMOS PARA LA LOCALIZACIÓN DE ABEJAS ROBÓTICAS Y EVITACIÓN DE COLISIONES

Juliana Henao Arroyave
Universidad Eafit
Colombia
jhenaoa4@eafit.edu.co

Gerónimo Zuluaga Londoño
Universidad Eafit
Colombia
gzuluagal@eafit.edu.co

RESUMEN

Para escribirlo pueden dar respuesta a estas preguntas:
¿Cuál es el problema?, ¿Por qué es importante el problema?, ¿Qué problemas relacionados hay?, ¿Cuál es la solución?, ¿cuáles los resultados? y, ¿Cuáles las conclusiones? Utilizar máximo 200 palabras.

Palabras clave

Abejas robóticas, colisiones, estructuras de datos

Palabras clave de la clasificación de la ACM

- Theory of computation→Design and analysis of algorithms
- Theory of computation→Data structures design and analysis
- Theory of computation→Algebraic complexity theory

1. INTRODUCCIÓN

Desde hace unos años ciertos factores que afectan el ambiente, como el clima, el uso de pesticidas, la destrucción de panales por tala de árboles, entre otros; ha causado que la población de abejas disminuya, lo cual tiene un impacto ambiental que claramente no solo las afecta a ellas, sino que su ida traería eventualmente la desaparición de muchas especies de plantas polinizadas por estos insectos que a su vez traen la desaparición de otras especies. Por esto es tan urgente hallar una solución alternativa a estos eventos que parecen irreversibles. Una manera es creando “abejas” robóticas que tomen el polen y lo lleven a otras plantas para completar la polinización. La creación de estas abejas trae consigo ciertas circunstancias que se deben controlar. En este caso trataremos el hecho de que las abejas podrían chocar unas con otras, la idea es usar estructuras de datos y diferentes algoritmos para hallar la solución para evitar que pasen este tipo de cosas.

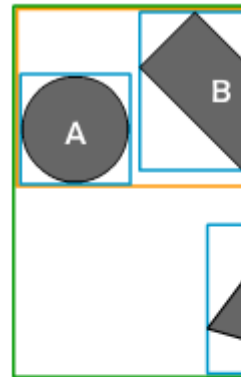
2. PROBLEMA

La creación de abejas robóticas (drones), trae consigo un obstáculo a la hora de la implementación de esta idea. Primero hay que considerar el numero de abejas que se tienen que utilizar para cubrir cierta área, con el objetivo de usar menos abejas y lograr cubrir un mayor espacio, sin embargo este no es el mayor problema al que nos enfrentamos al momento de crear estos drones. Teniendo en cuenta lo necesario que es este proceso natural de polinización, es muy claro que va a ser necesario la creación de un numero muy grande de estas abejas robóticas, la consecuencia real de esto es al momento de evitar la colisión entre ellas. Hay que entender que estas abejas son “autómatas” y que por razones de cantidad, no

Aquí deberán explicar que se encuentran científicos o sitios web de ellos. NO poner solo

3.1 Dynamic AABB tr

La estructura de datos para la detección de colisión que hace uso de la estructura de datos de árbol de búsqueda binaria aplicado recursivamente. Si un objeto no tiene un hijo próximo o tiene una intersección con un hijo, de esta manera, se puede determinar si los objetos o incluso es una colisión. Por ejemplo el impacto de la colisión de los objetos o estructuras de un juego. En aplicaciones, hasta en la predicción de colisiones que tiene aplicaciones en la simulación hasta en la simulación



debe definir ciertos parámetros para evitar colisiones o congestión entre estos drones. Ante esto el Laboratorio de Sistemas Inteligentes de Stanford (SISL) busca crear un sistema para gestionar este tráfico aéreo no tripulado. Se utilizaron varias soluciones, pero la densidad esperada de los vuelos de drones creó un nivel completamente nuevo de complejidad para el equipo de SISL. "A medida que el número de aviones crece, el problema de evitación se vuelve exponencialmente más complicado, un desafío que los matemáticos llaman la maldición de la dimensionalidad", dijo el estudiante graduado de ingeniería mecánica Hao Yi Ong, que detalla un algoritmo de prevención de conflictos. "Así que tenemos que encontrar mejores formas que la simple búsqueda de fuerza bruta e iterar a través de todas las soluciones posibles".

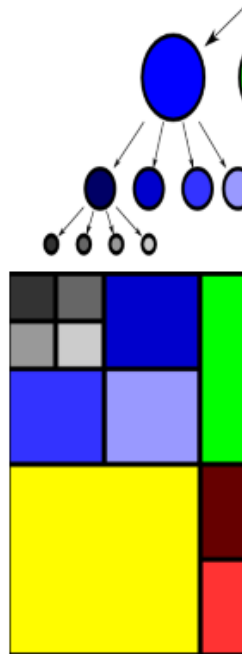
La solución que propuso Ong para este problema fue separar los conflictos de aviones múltiples en problemas emparejados. Y rápidamente seleccionar la mejor acción para cada par de drones desde una tabla que predice la trayectoria de vuelo de cada dron. El servidor luego coordina cada una de estas soluciones por pares y emite una orden conjunta de prevención de colisiones para todos los drones afectados.

3.3 Hashing

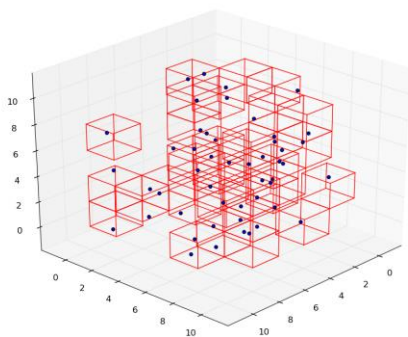
Hashing implica aplicar un algoritmo hash a un elemento de datos, conocido como la clave hash, para crear un valor hash. Los algoritmos hash toman una amplia gama de valores y los asignan a un conjunto de valores más pequeño. Los valores hash se pueden usar para acelerar la recuperación de datos y se pueden usar para verificar la validez de los datos. Una coalición es cuando dos o más teclas hash resultan en el mismo valor hash. Esto causa un problema ya que ya no podemos encontrar rápidamente si los datos están en nuestra tabla hash o no, ya que otra pieza de datos puede tener el mismo valor. La solución a este problema se puede dar de dos maneras, una es colocando las claves colisionantes en el siguiente valor hash libre

3.4 Quadrees y Octrees

Los Quadrees (en 2D) y Octrees (en 3D) son estructuras de datos en las que cada nodo tiene a su vez cuatro nodos hijos. En nuestro escenario en 2D, cada nodo en esta estructura de datos representa un objeto en cada interacción y un objeto está colisionando



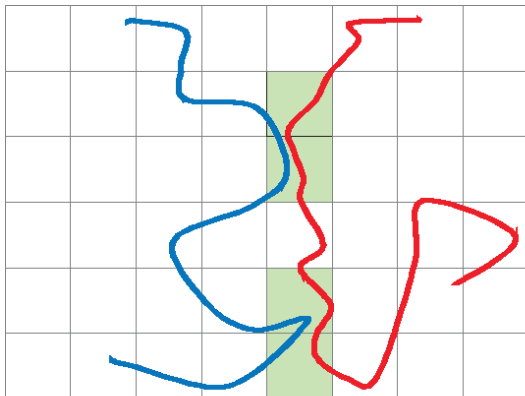
4. Spatial Hashing



Gráfica 1: Visualización del spatial hashing en 3D.
(Imagen sacada de: <http://www.sgh1.net/posts/spatial-hashing-1.md>)

4.1 Operaciones de la estructura de datos

Para el funcionamiento de estas tablas hash se pueden implementar varias operaciones como agregar un casillero, buscar uno en específico, borrar uno y hallar colisiones.



Gráfica 2: Imagen de una colisión en una tabla hash en 2D
(Imagen sacada de: http://www.computationalgeometry.com/notes/Parallel_Spatial_Hash_Tables)

REFERENCIAS

1. Chipman, Ian. St. predict and prevent dr. Magazine. December. <https://engineering.star-am-develops-software->
2. https://en.wikibooks.org/wiki/Fundamentals_of_data_structures/Hash_tables_and_hashing
3. https://upload.wikimedia.org/wikipedia/commons/2C/2C_graphical_hashing.png
4. https://es.wikipedia.org/wiki/Tabla_hash
5. content/uploads/2014/06/Tabla-hash.png
6. <http://www.mastergrafica.com/2015/06/deteccioncolision/>