

Control y supervisión algorítmica para la prevención de colisiones

Hamilton Smith Gómez Osorio

Santiago Isaza Cadavid

Medellín, 6 de noviembre del 2018

Estructuras de Datos Diseñada

Matriz Tridimensional

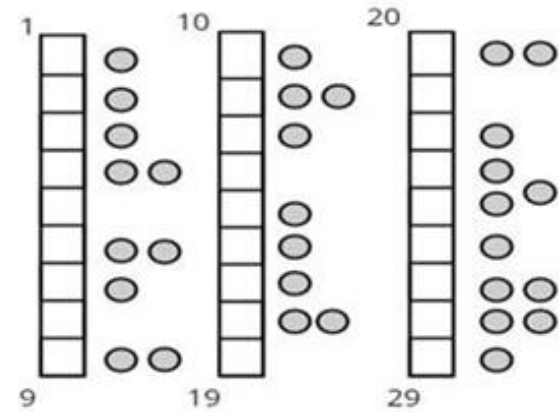
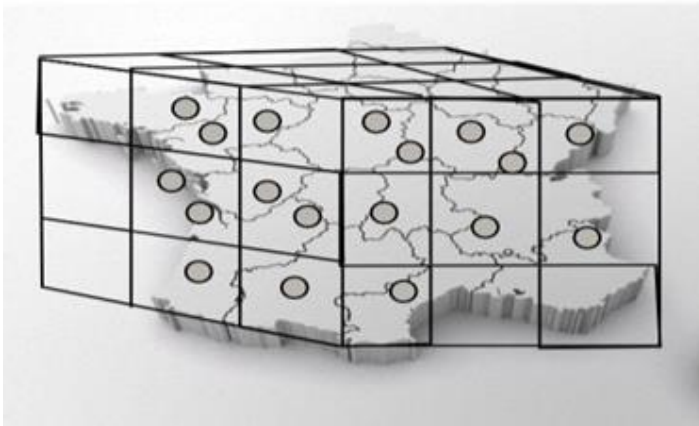


Gráfico 1-2: representación de la matriz en el mapa de Bello y clasificación de abejas

Operaciones de la Estructura de Datos

- **Crear Celdas:** Identificando los valores máximos y mínimos de cada coordenada construimos celdas de diagonal 100 y lados 57,7.
- **Detectar colisión:** Clasificamos cada abeja respecto a su posición y las ingresamos a una pila.
- **Abejas adyacentes:** cuando una abeja está sola en una celda buscamos en las celdas alrededor de esta ya que podrían colisionar.

Método de la estructura	Complejidad
areaDeUbicacion()	$O(n)$
abejasAdyacentes()	$O(m)$
detectarColisiones()	$O(n + m)$

Tabla 1: Complejidad de las operaciones de la estructura de datos

Criterios de Diseño de la Estructura de Datos

- La complejidad debe ser menor a $O(n^2)$
- Insertar y borrar en una pila es $O(1)$
- La matriz tridimensional representa a su vez un espacio.
- Desde que dos abejas colisionen, no deben analizarse con otras.

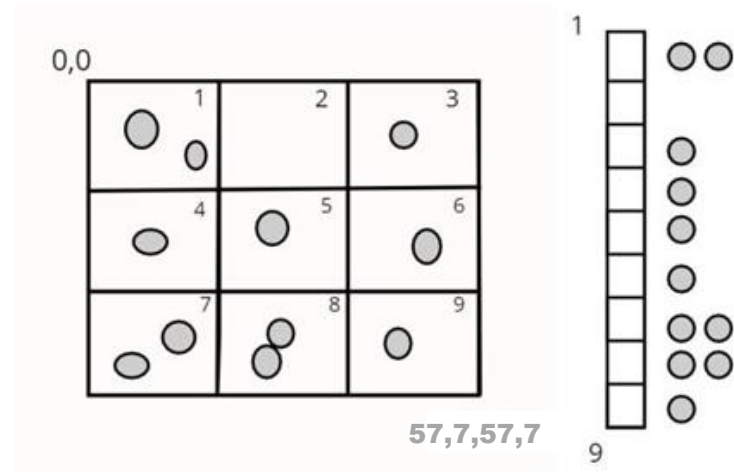


Grafico 3: visión bidimensional del problema

Consumo de Tiempo y Memoria

Método	Tiempo promedio(ms)						
	Número de abejas						
	4	10	100	1.000	10.000	100.000	1'000.000
areaDeUbicacion()	0,6	3,6	3,6	11,6	28,4	275,8	3.315,2
detectarColisiones()	0	4,2	11	8,2	37,8	131,8	673,6
guardarArchivo()	1,6	3,2	7,4	5	112,8	770	6.060,2

Número de abejas	Memoria
10	1
100	2
1000	4
10000	15
100000	80
1000000	245

Tabla 2-3: Tiempo de ejecución y consumo de memoria para los diferentes tamaños del problema

Software Desarrollado

Número de abejas	Resultado
4	4
10	4
100	24
1000	279
10000	9546
100000	99184
1000000	1000000

Tabla 4: resultados obtenidos para cada tamaño del problema

$$X = \frac{(Max - Min) * 111111}{\frac{100}{\sqrt{3}}}$$

Ecuación 1: tamaño del eje coordenado

$$D = \frac{(x - Min) * 111111}{\frac{100}{\sqrt{3}}}$$

Ecuación 2: clasificación de abejas

REFERENCIAS

- [1]. Liu, B. and Su, X. An Anti-Collision Algorithm for RFID Based on an Array and Encoding Scheme. Information, 2018, 2078-2489. Accessed August 25, 2018 from Universidad EAFIT: <https://bit.ly/2PEzPhx>
- [2]. Dinas, S. and Bañón J. M. A literature review of bounding volumes hierarchy focused on collision detection. Ingeniería Competitiva, 2015, 49-62. Accessed August 25, 2018, from Universidad EAFIT: <https://bit.ly/2BMI9sD>
- [3]. Nevala, E. Introduction to Octrees. GameDev.net, 2018. Accessed September 23, 2018: <https://bit.ly/2pxAzJa>
- [4]. Spatial hashing implementation for fast 2D collisions. The mind of Conkerjo, 2013. Accessed September 23, 2018: <https://bit.ly/2xHprNK>
- [4]. How to efficiently remove duplicate collision pairs in spatial hash grid? Stack Overflow, 2015. Accessed September 23, 2018: <https://bit.ly/2O2PPvG>