

Reporte Tarea 3a: “Acuario”

Nombre: Lucas Oyarzún

Fecha: 18 de julio de 2020

Curso: CC3501 - Modelación y Computación Gráfica para ingenieros

Profesor: Daniel Calderón

Solución Propuesta:

Se crea programa “aquarium_solver.py”, que dados ciertos parámetros en archivo “problema-setup.json”, resuelve la ecuación de calor para un acuario con 2 calentadores en su parte inferior. Este programa crea el archivo “solution.npy” y “datos_pecera.npy” con la matriz de solución en 3D del sistema de ecuaciones diferenciales y los datos del acuario. Además, este programa muestra una representación en 3D del espacio del acuario con matplotlib (Se ven los colores de temperatura de forma precisa con matrices de $N \times N \times N$ dimensiones).

Se continúa creando el programa “peces.py” Donde hay 3 funciones que crearán las Shapes de 3 peces diferentes, que llamaremos “Nemos”, “Doris” y “Espadas”, este programa, al ser arrancado, muestra una visualización de los 3 tipos de peces y su movimiento de cola.

También se crea el programa “funciones_marching_cubes.py” con las funciones de Marching Cubes entregadas en el Auxiliar 8, con ciertas modificaciones para este trabajo.

Finalmente, el programa principal es “aquarium-view.py”, que, ocupando la matriz “solution.npy” y el archivo “view-setup.json”(con datos de temperaturas favoritas y cantidad de peces de 3 tipos), muestra una visualización del acuario deseado.

Para esto, comienza creando una Shape de los bordes de la pecera, luego, se crea una malla 3D que usaremos como espacio para el resto de funciones. Creamos los voxels de colores rojo(t_a), amarillo(t_b) y azul(t_c), que representarán los voxels con las temperaturas indicadas en “view-setup.json”, con un margen de $\pm 2^\circ$.

Al momento de crear los voxels, guardamos sus posiciones en una lista, de la cual extraeremos(extraemos para que no se repitan(superpongan), si solo la leyéramos, se podrían superponer los peces) aleatoriamente, y creamos la cantidad (n_a, n_b, n_c) de peces indicada en “view-setup.json” de cada tipo, en las posiciones extraídas y con un giro aleatorio de hacia dónde ven.

Se configura para que con números cambiemos el tipo de visualización, con las flechas y W, S, movemos la cámara y con A, B, C, la temperatura mostrada.

Los voxels son dibujados con el algoritmo de Z-Buffer, lo que produce que se vean más transparentes por un lado del acuario. Esto podría ser interpretado como un error, pero personalmente, lo considero una ventaja, de tal forma que se ve como “por dentro” por un lado(se notan más los peces) y “por fuera” por el otro lado del acuario (se nota más el agua), por lo que se decide dejar de esta forma la visualización.

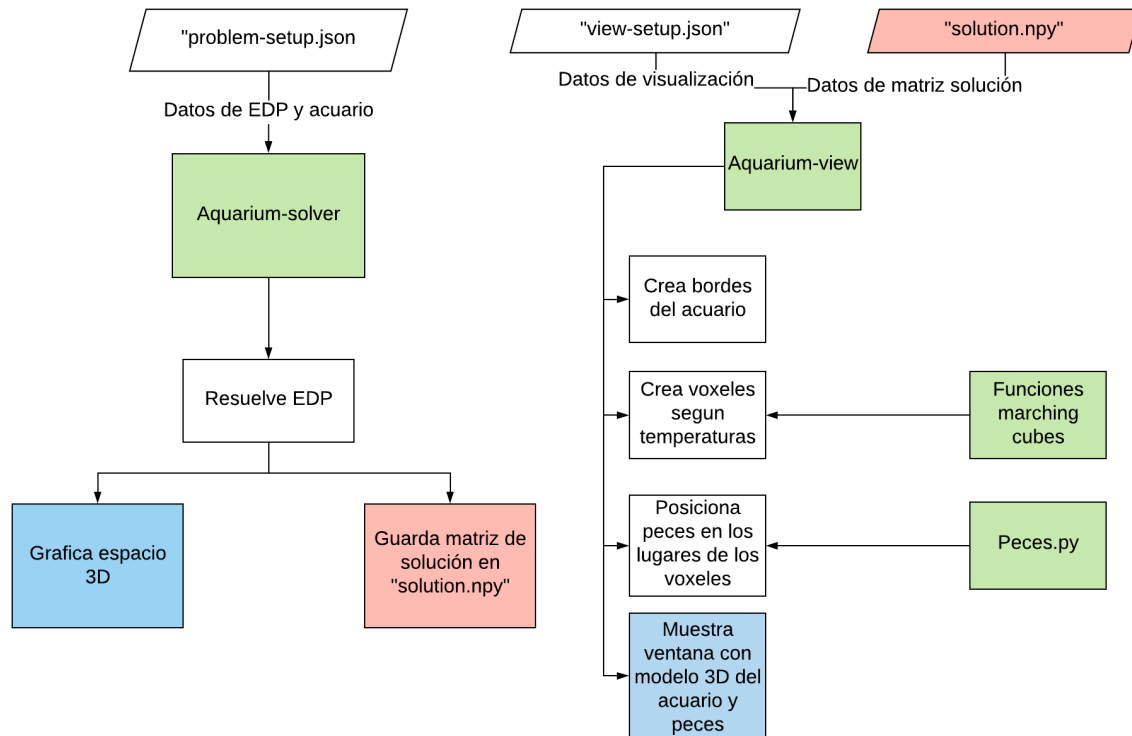


Diagrama de soporte del programa

Instrucciones de ejecución:

Argumentos recibidos: archivos “problem-setup.json” (Datos de la pecera y sus condiciones de borda) y “view-setup.json”(Cantidad de peces y temperatura preferida).

Forma de ejecución: con llamadas: `python aquarium-solver.py problem-setup.json`
`python aquarium-view.py view-setup.json`

Teclas de control: W,A: Acercan y alejan la cámara del acuario. Flechas del teclado: Rotan la cámara alrededor del acuario. A, B, C: muestran/ocultan los voxeles de t_a , t_b , t_c respectivamente. 1: Muestra voxeles transparentes. 2: Muestra voxeles sin transparencia. 3: Oculta agua para apreciar mejor los voxeles de las temperaturas.

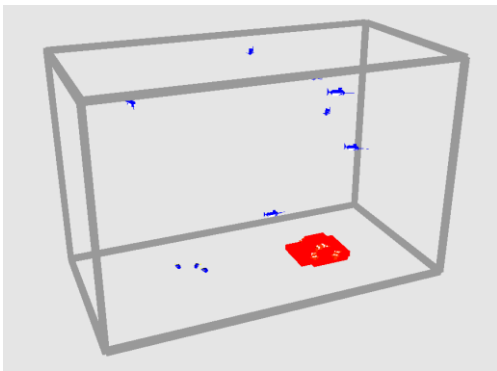
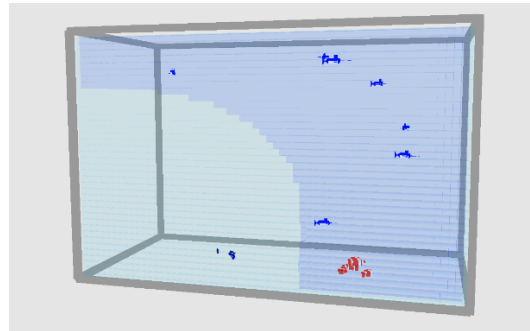
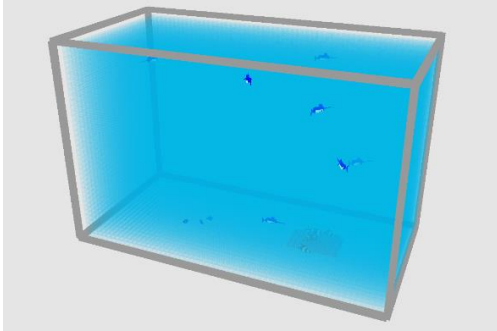
Control: Podemos ver desde diferentes ángulos el acuario con W, A y las flechas del teclado. Si apretamos 3, podemos hacer desaparecer el agua, y si apretamos 1 o 2, modificamos la transparencia del agua (desactivar el agua al apretar 2 para apreciar correctamente). Además de A, B, C, para visualización de temperaturas

Al crearse la ventana, se crean todos los objetos y se posicionan, dándoles un movimiento continuo a las colas de los peces. Lo que se modifica durante la ejecución es el ángulo de vista de la cámara y su distancia al acuario, además de la forma de dibujar los voxeles(y la decisión de dibujarlos o no).

Dependiendo de si el acuario se ve de frente o desde atrás, aumenta o disminuye la transparencia del agua (Esto no se modifica porque se considera una forma de ver “dentro del agua” muy estética).

Screenshots:

1. Imágenes del acuario



2. Visión cercana del modelo de Nemo

