

Low-hanging fruit

El código que acompaña es una simulación sencilla de dinámica de fluidos en tiempo real utilizando las ecuaciones de Navier-Stokes [^1].

Al correr `make` se crean dos ejecutables, `demo` y `headless`:

- `demo` grafica el estado de la simulación a una ventana y permite interactuar con el modelo. Requiere instalados paquetes de desarrollo para OpenGL, GLU, y GLUT (`apt install mesa-common-dev libglu-mesa-dev freeglut-dev`) para poder compilar.
- `headless` no tiene interfaz con el usuario y está hecho para medir la performance de la simulación. Durante la ejecución imprime periódicamente los nanosegundos que tomó actualizar cada elemento de la matriz de estado en promedio.

Ejercicios

1. Compile el programa con la versión más nueva de GCC que tenga disponible, agregando de a uno los distintos flags de compilación vistos.
- LTO es de gran importancia en esta simulación. ¿Por qué?
2. Compare el mejor resultado que obtuvo con los de la versión más vieja de GCC que pueda instalar, utilizando flags similares.
3. Repita el experimento utilizando Clang o ICC y compare.
4. Este código ya ha sido optimizado en parte, cambiando el método de Gauss-Seidel tradicional de `lin_solve` en `solver.c` por red-black Gauss-Seidel, donde las actualizaciones de cada color son independientes. Además, la disposición de la matriz en memoria se ha cambiado de modo que las celdas negras y rojas estén contiguas en memoria. Al recorrer elementos independientes que están contiguos en memoria, el compilador puede autovectorizar el código. Corra el código original antes de esta optimización y compare tiempos.

Bibliografía

[^1] Jos Stam, *Real-Time Fluid Dynamics for Games*, GDC 2003.