



VNIVERSITAT
DE VALÈNCIA

Escola Tècnica Superior
d'Enginyeria **ETSE-UV** 

Práctica 3

Simulación

3º Ingeniería Multimedia

Profesor:

Miguel Lozano Ibáñez

Estudiantes:

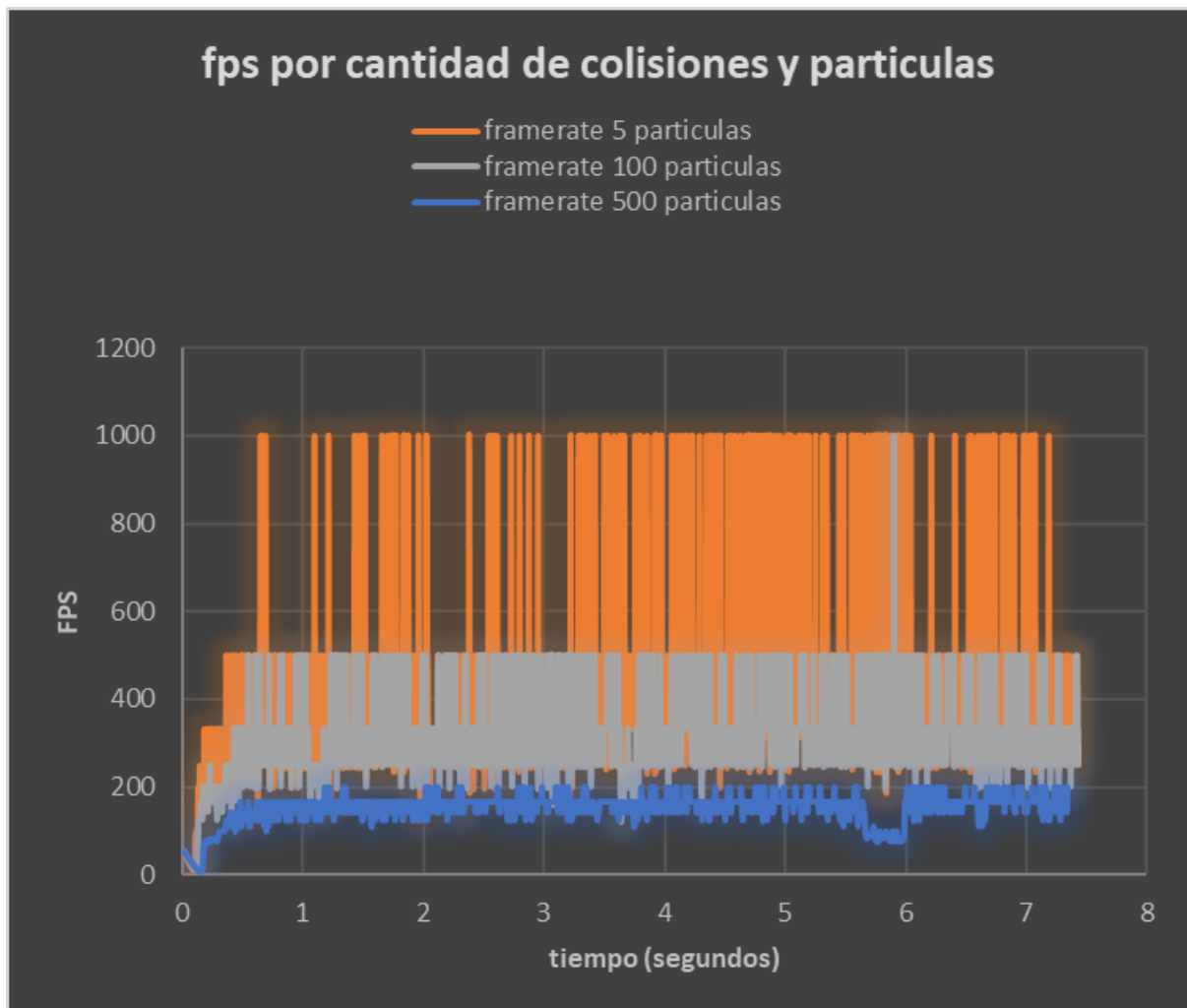
Óscar Marín

Francisco Sevillano

FECHA: 02/05/2022

Descripción del problema 1 (mesa de billar)

Se busca implementar un modelo de partículas que colisionan entre sí y diversos planos siguiendo diferentes métodos de colisiones. Para la primera parte se busca realizar las colisiones que sigue una mesa de billar (colisiones entre los lados y las partículas) el modelo usado para las colisiones entre partículas es un modelo basado en velocidades donde se aplica un concepto bastante similar al de las paredes.



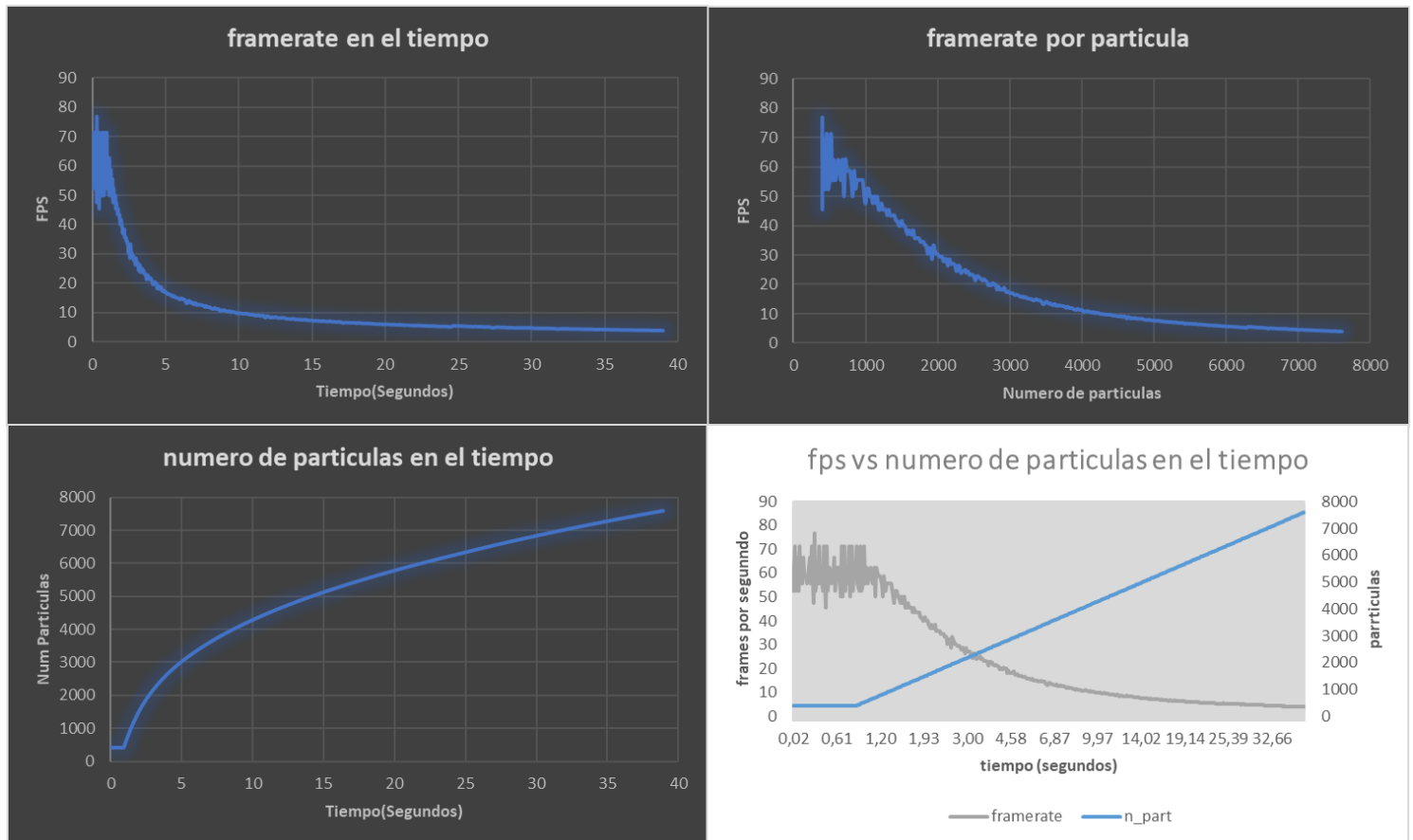
5 part - 1000 alcanzados : 100 part - 500 fps alcanzados : 500 part - 144 fps

Como se puede observar en la gráfica la escalabilidad del sistema es buena pero ineficiente como veremos más tarde en nuestro caso ha hecho falta llenar la mesa de partículas para que el rendimiento se vea afectado.

Las colisiones entre partículas de esta manera son costosas debido a que cada una debe comprobar el resto y en caso de interactuar con alguna realizar diversos cálculos para llevar a cabo la colisión. Esto puede suponer costes de orden de n^2

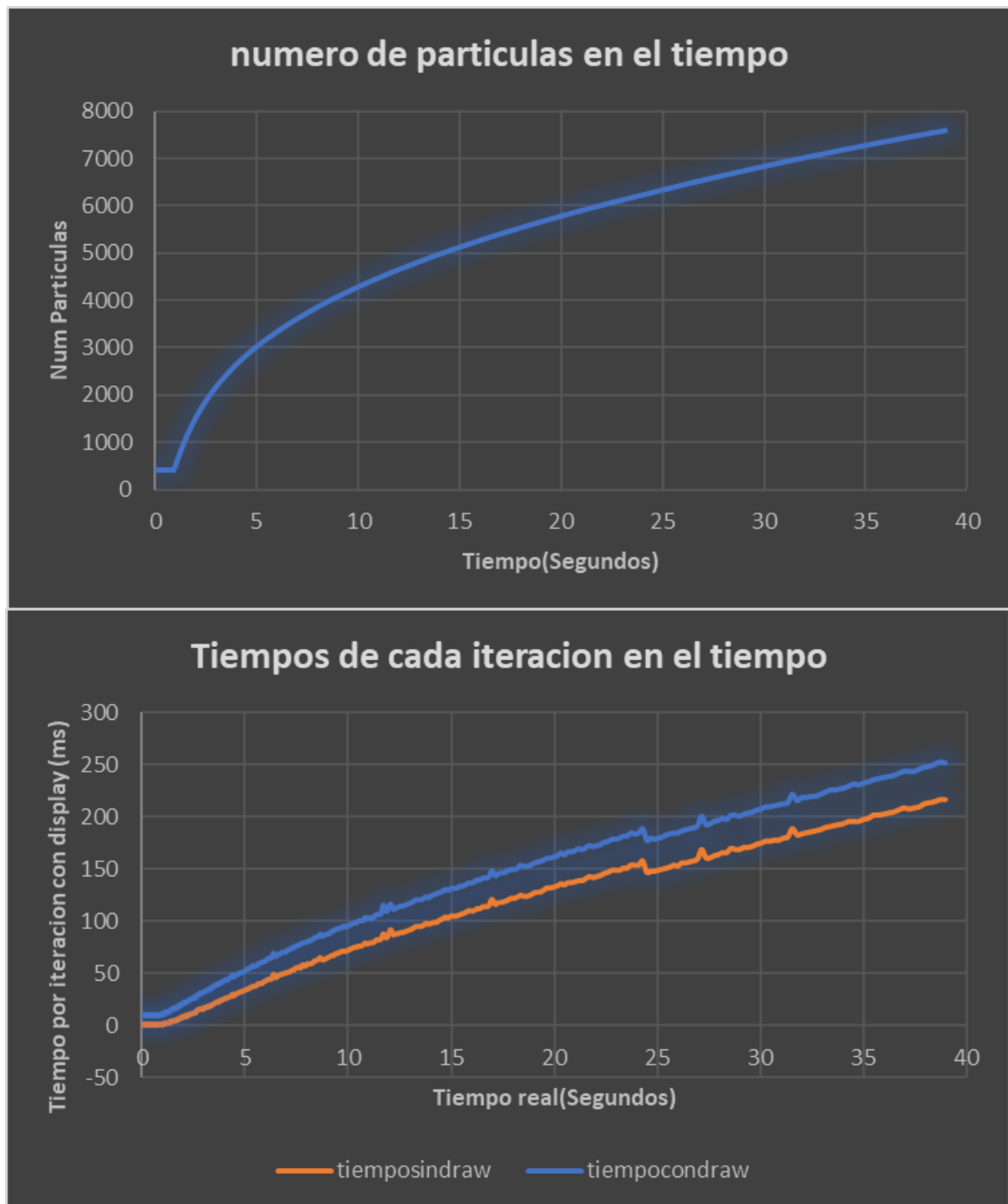
Partículas y Estructura de datos

Para esta segunda parte se pide que se genere un sistema de partículas con colisiones elásticas entre ellas y metidas en un recipiente con colisión con las paredes para mantenerlas dentro. Cada partícula comprobará la colisión con cada una de sus hermanas, una cosa no muy eficiente, por este motivo más tarde aplicaremos estructuras de datos para que solo compruebe las partículas cercanas.



Como podemos observar, conforme se aumentan las partículas el framerate decae drásticamente dado que tiene que comprobar cada una de estas con todas las demás, esto se hace a través de un doble for uno por cada partícula y otro para comprobar cada partícula por partícula por lo que no es de extrañar que el decaigan los fps de forma cuadrática, en las gráficas no se refleja esta manera y puede ser debido a varios factores como pueden ser que se tengan en consideración más cosas a parte de ese bucle anidado.

Parte 2: Partículas y Estructura de datos (Colisiones y Partículas)



El de estas gráficas podemos observar como también va aumentando lo que le cuesta dibujar las partículas contra mas hay pero también podemos observar como el dibujado es un coste pequeño en comparación con el cálculo

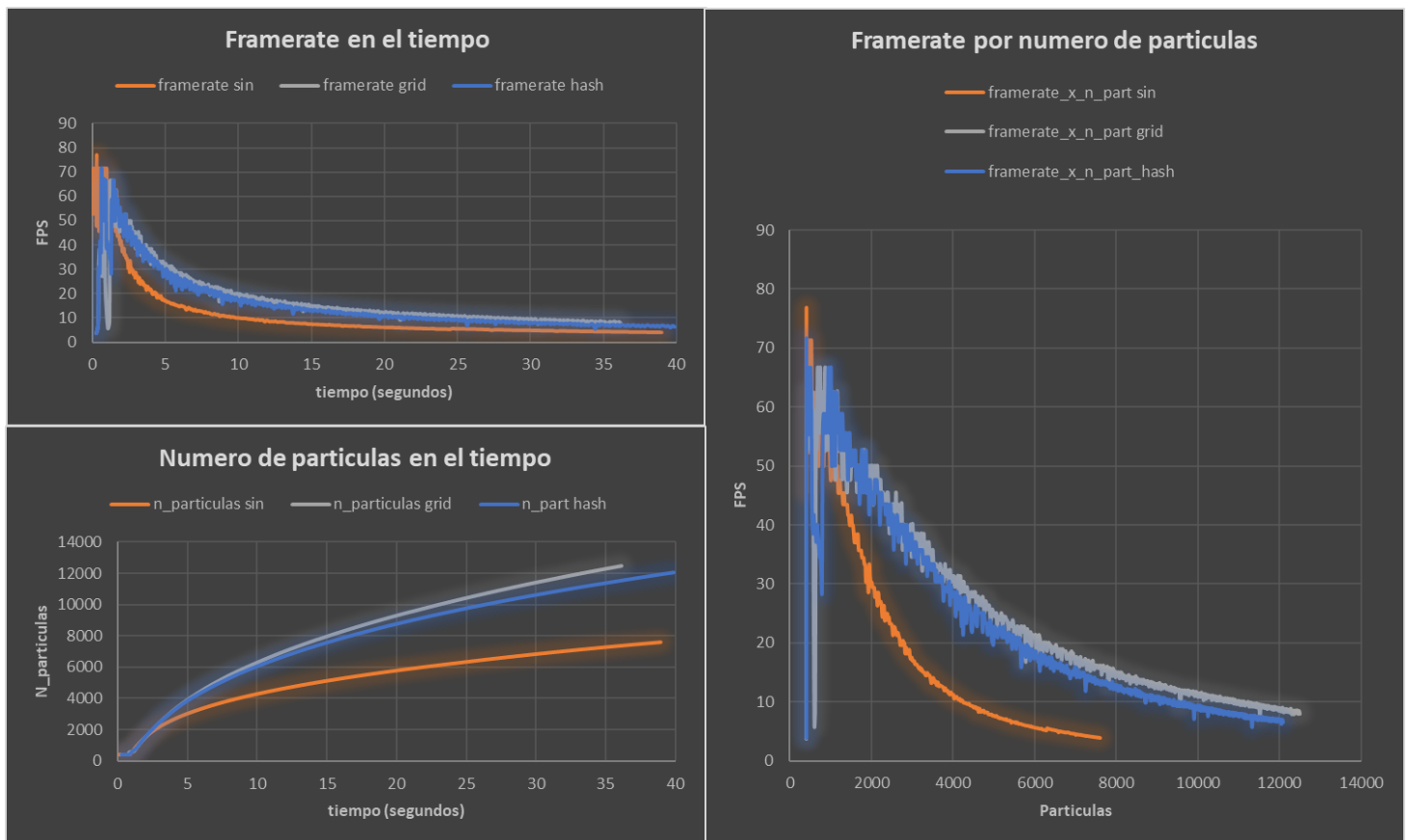
Parte 3: Partículas y Estructura de datos (Estructuras)

Las estructuras de control utilizadas son:

Grid que consta de como su nombre indica una rejilla donde se almacenan partículas en cada uno de sus huecos dependiendo de su posición

Hash que se basa en el mismo concepto pero se trata de reducir la memoria utilizada al almacenar varias celdas dentro de la misma siguiendo un algoritmo de hashing.

Ambas se basan en el mismo concepto de guardar partículas y buscar las vecinas para realizar las colisiones, en nuestro caso lo hemos realizado con 4 vecinas y 30 divisiones horizontales y verticales.



Como podemos observar en las gráficas la mejora en cuanto rendimiento al aplicar estructuras es notable, las diferencias entre ambas no son muy apreciables y aunque el grid sobresalga un poco por encima puede ser debido a que al haber muchas partículas y estar por todas partes entra en el terreno de ventaja del grid. Si las partículas estuvieran más quietas el hash saldría beneficiado.