

# Simulación de carácter catastrófica del Sistema Solar en OpenGL

Wladimir Albornoz, *estudiante, Usach* y Andrés Barrera, *estudiante, Usach*

**Resumen**—En el siguiente informe se detallan los aspectos preliminares para el proyecto de la asignatura de computación Gráfica, en este se presenta el modelamiento de sistema solar con un cinturón de asteroides, todo esto en un sistema de software. Otras implementaciones realizadas en este trabajo fueron la aplicación de texturas en el fondo de la escena utilizando la técnica de SkyBox y la implementación de un menú para facilitar la navegación entre las funcionalidades del programa y el experimento. Fue implementado en OpenGL bajo el sistema operativo GNU/Linux.

**Index Terms**—Sistema solar, OpenGL, rotación, traslación, objetivos, herramientas a utilizar, plan.



## 1. INTRODUCCIÓN

LA idea principal del proyecto es generar una simulación de lo que pasaría si el Sol absorbiera a los planetas del sistema solar, fundamentalmente si se rompiera la resistencia existente, debido a que el sol se transformará en una gigante roja esto sucederá dentro de 10 mil millones de años aproximadamente [1].

Esta simulación tiene un carácter científico, ya que está demostrado que cuando se concrete el fin del sol terminaría absorbiendo a los planetas que se encuentren cercanos a él, se toman como base las diferentes visiones e información que se tiene del espacio-tiempo.

Debe considerarse que lo que se está presentando toma como base parte de un proyecto ya implementado [2], cabe destacar que esto resulta mas complejo debido a que la base está realizado con aspectos, métodos, módulos y herramientas diferentes, teniendo que considerar un tiempo extra para injertar de manera perfecta esto a el proyecto que se está realizando.

## 2. OBJETIVOS

### 2.1. Objetivo General

Implementar gráficamente la simulación catastrófica del sistema solar y que se aprecie lo mas apegado a la realidad posible.

### 2.2. Objetivos Específicos

- Conocer y relacionar términos de la computación gráfica para el desarrollo del proyecto [3].
- Entender y aprender a utilizar OpenGL para la implementación y representación de elementos de la computación gráfica.
- Determinación de herramientas útiles para el desarrollo.
- Descubir librerías que puedan ayudar tanto en la codificación como en la implementación.
- Lograr dominar el lenguaje y las herramientas a utilizar.

## 3. DESCRIPCIÓN DE LA PROBLEMÁTICA

### 3.1. Motivación

Resulta muy interesante la oportunidad de generar una simulación gráfica el sistema solar [4] conocido (figura 1), mas detalladamente del fin de el sol como se conoce, debido a la importancia que este sistema tiene para toda la humanidad; la mayor motivación para el desarrollo es en efecto poder implementar el proyecto, además con ello se pretende aprender y a la vez poner en práctica conceptos de la computación gráfica como la rotación, traslación y el modelado en 3 dimensiones.

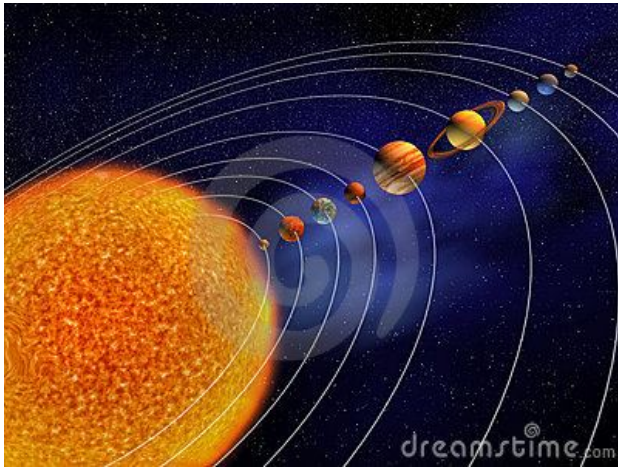


Figura 1: Sistema solar conocido

### 3.2. Definición de la problemática

La mayoría de las implementaciones gráficas conocidas muestran simulaciones de la realidad de los planetas y del sistema solar, sus trayectorias, sus lunas o las diferencias entre las velocidades de rotación de cada uno; pero ¿qué sucedería si las órbitas de los planetas aumentara y sumados al aumento de la masa del sol, este atrajera a todos los planetas existentes en el sistema conocido?, el fin de el sol y de los planetas como los conocemos es lo que se desea implementar.

Skybox es una técnica que permite que una escena se vea más grande y más impresionante, envolviendo al usuario con una textura que es posible apreciar en una cámara de 360°[5].

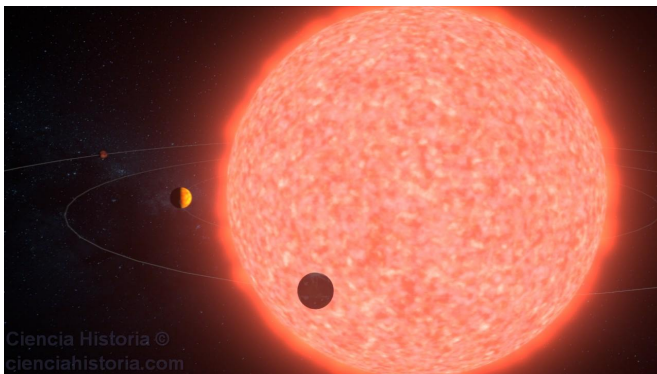


Figura 2: El Sol convertido en gigante roja llegando casi a la órbita de Venus, la Tierra tiene un futuro incierto

## 4. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

Tomando como base el proyecto que solo muestra una representación del sistema solar, se pretende que la representación de las trayectorias de los planetas sea automática, con esto se refiere a que se vea el movimiento continuo de la trayectoria de los planetas y que eventualmente de a poco los planetas comiencen a cambiar su trayectoria, los cuales sumados al aumento del diámetro del sol, como se muestra en 2, empiecen a ser absorbidos por este, lo cual los hará desaparecer[6].

Para ello habría que:

- Modificar algunas funciones relacionadas con la traslación de estas diferentes figuras (diferentes planetas)
- Editar y acondicionar las formulas de elipse.
- Aumentar el radio del sol.

## 5. MODELAMIENTO

### 5.1. Función de la elipse

A continuación en la figura 3 se puede ver una elipse para posteriormente analizar su función matemática:

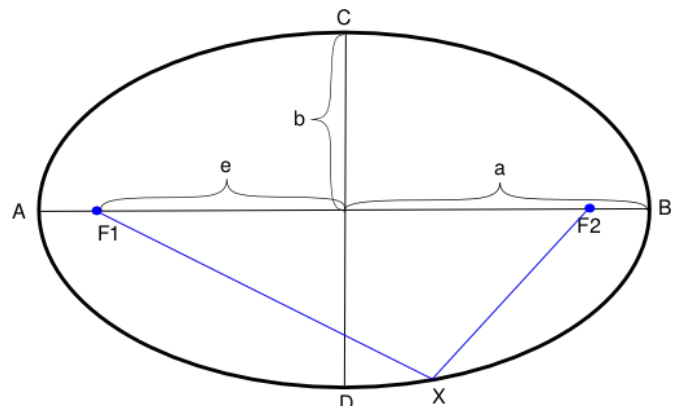


Figura 3: Elipse

Se utiliza la función de la elipse:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2}$$

Donde  $a > 0$  y  $b > 0$ , las coordenadas de las abscisas. La elipse se encuentra centrada en el origen ( $h, k = 0$ ). La ecuación de la elipse modela la ubicación de la órbita de los planetas.

## 5.2. Transformaciones 3d

Se utilizan una serie de transformaciones en el desarrollo del problema, estas serán mostradas a continuación:

### 5.2.1. Rotación

Claramente como se desea implementar un sistema solar, un detalle importante a destacar es la rotación que tiene cada planeta; la rotación está definida por la siguiente matriz:

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

En vista de que se trabaja en tres dimensiones es necesario definir las matrices para cada tipo de rotación que se pueden obtener desde la matriz de rotación general de 2x2:

Rotación sobre el eje x:

$$R_{xy}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotación sobre el eje y:

$$R_{yz}(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotación sobre el eje z:

$$R_{zx}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Se describe como rotación en los planos  $xy$ ,  $yz$  y  $zx$ , pues resulta más fácil asociar la rotación en 3 dimensiones de esa forma. La órbita de los planetas se genera aplicando una rotación sobre el eje  $y$  (En el origen, teniendo el solo como referencia) con un ángulo  $0^\circ \leq \theta \leq 360^\circ$ , el cual aumenta en cada frame de la escena.

### 5.2.2. Traslación

La matriz que describe la traslación de los objetos en la escena es la siguiente:

$$T = \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} v_x \\ v_y \\ v_z \\ v_w \end{bmatrix} = (v'_x, v'_y, v'_z)$$

La traslación permite saber el radio que tendrá la órbita del planeta respecto al sol. Es en torno a esta transformación que se aplica la rotación, por lo tanto para cada frame de la escena se aplica un pequeño aumento en la rotación respectiva.

## 5.3. Distancia entre dos puntos

Esta fórmula se utiliza para determinar la distancia entre 2 puntos en un plano en 2D, aunque el sistema está representado en 3 dimensiones. Los planetas se mueven respecto al eje  $y$  y la posición de los mismos está determinada sólo por sus coordenadas en el eje  $x$  y en el eje  $z$ :

$$x_{new} = \frac{\sqrt{X_{final}^2 + X_{inicial}^2}}{100}$$

$$z_{new} = \frac{\sqrt{Z_{final}^2 + Z_{inicial}^2}}{100}$$

la distancia total entre el punto final y el inicial es dividida por un número random con ello se genera la traslación de un objeto desde un punto hacia otro; lo que se traslada es la viewport

## 6. DISEÑO DEL PROBLEMA

### 6.1. Diseño preliminar

### 6.2. Diseño detallado

## 7. METODOLOGÍA Y HERRAMIENTAS A UTILIZAR

### 7.1. Metodología

La metodología a utilizar es el análisis y diseño estructurado ya que con esta metodología se tiene un orden lógico en los pasos a seguir para el desarrollo correcto del proyecto.

## 7.2. Herramientas

Preliminarmente se tienen las siguientes herramientas para el desarrollo:

### 7.2.1. Equipos

Se posee 2 notebooks con similares características, procesadores Intel core I3 y I5, CPU de 2.4 GHz, 8 Gb de memoria Ram y distribución de Linux Ubuntu 14.04 LTS.

### 7.2.2. Herramientas de programación

Preliminarmente se establecen las siguientes herramientas para la programación, tales como:

- Netbeans IDE 8.0.2  
Entre algunas cosas destaca la rápida observación de errores sintácticos y fácil compilación.
- OpenGL  
Principal entorno de desarrollo, para la creación de aplicaciones gráficas portátiles e interactivas en 2D y 3D [7].

## 8. ANÁLISIS DE RESULTADOS

### 8.1. Validación de resultados

Como se especificó en la presentación preliminar de la idea, para realizar esta aplicación se toma como base un proyecto anterior del ramo de computación gráfica, el cual se llama Representación del sistema solar en OpenGL". De este código se toman las funciones principales para generar la representación y se eliminan funciones que no son utilizadas como las que representan estrellas específicas, etc. En el código se representa Plutón que no se encontraba representado y además se comienza generando todos los planetas alineados tras el Sol para mantener un orden más lógico en un principio.

### 8.2. Rendimiento preliminar

En esta etapa de prototipado del proyecto, el equipo de trabajo se enfoca en implementar las funciones más importantes del proyecto base además de aprender sobre la utilización de la herramienta OpenGL. A la base tomada para el desarrollo se le eliminan una serie de funciones que en realidad no serán necesarias por lo tanto esto entregará un código más

compacto y fácil de comprender.

Preliminarmente no se identifican problemas relacionados a la presentación en pantalla de la aplicación al ser ejecutada, la velocidad de movimiento (rotación) es más que suficiente para identificar los movimientos de los planetas.

### 8.3. Problemáticas y funciones faltantes

## 9. PLAN DE TRABAJO

Se realiza una carta Gantt para darle fechas límites a cada tarea que este proyecto conlleva. Se adjunta para no entorpecer el modelo de este documento y para que la carta Gantt se pueda analizar de mejor manera.

## 10. CONCLUSIONES

Como inicio para un buen proyecto, siempre se deben llevar a cabo investigaciones que puedan determinar cual tema abordar y que herramientas utilizar para un óptimo desempeño durante este.

El tema abordado nos llena de optimismo para trabajar en él, sumado a los conceptos de computación gráfica aprendidos más la investigación pertinente, nos motiva a dar el 100 % de nuestro esfuerzo para hacer de este un gran proyecto.

## REFERENCIAS

- [1] R. G. Baker, "The sun-earth connect 1: A fractional -matrix of solar emissions compared to spectral analysis evidence of solar measurements and climate proxies," 2015.
- [2] S. P. J. Riveros, "Sistema solar," 2008.
- [3] J. Foley, *Computer graphics : principles and practice*. Reading, Mass: Addison-Wesley, 1995.
- [4] Astromia.com. (Diciembre 15, 2014.) Astronomía educativa: Tierra, sistema solar y universo. [Online]. Available: <http://www.astromia.com/solar/nubeoort.htm>
- [5] Atspace. (Diciembre 15, 2014.) Tutorial 25 - skybox. <http://ogldev.atspace.co.uk/www/tutorial25/tutorial25.html>.
- [6] E. Siegel. (Enero 2, 2014.) The far future of our solar system. <https://medium.com/starts-with-a-bang/the-far-future-of-our-solar-system-31d05b036596>.
- [7] OpenGL. (Diciembre 15, 2014.) OpenGL overview. [Online]. Available: <https://www.opengl.org/about/>