

El problema de los 2 cuerpos y el caos en sistemas de 3 o más cuerpos

Proyecto del curso CA0305 - Herramientas de Ciencia de Datos II

Jose Andrey Prado Rojas C36174^{*1}, Debbie Con Ortega C32250^{*2}, Jesús Aarón Retana Castro C26400^{*3}

**Estudiantes de Ciencias Actuariales Pura, Escuela de Matemática, Universidad de Costa Rica. San José, Costa Rica. Junio, 2025*

¹joseandrey.prado@ucr.ac.cr, ²debbie.con@ucr.ac.cr, ³jesus.retana@ucr.ac.cr

Introducción

El movimiento de los cuerpos celestes ha despertado un profundo interés durante siglos, tanto para la comunidad física como matemática. De hecho, ha sido impulsor de revoluciones científicas y filosóficas a lo largo de la historia. En sus inicios, los grandes pensadores centraban sus ideas en el movimiento de los cuerpos del sistema solar.

A lo largo del tiempo la discusión sobre el movimiento de los cuerpos celestes ha persistido, dando lugar a una variedad de teorías que en ocasiones se han mezclado y coexistido entre ellas. Desde la Teoría Geocéntrica formulada por Aristóteles y desarrollada por Ptolomeo, hasta la Teoría Heliocéntrica propuesta por Nicolás Copérnico en el siglo XVI y posteriormente confirmada por Galileo Galilei. Finalmente, la comunidad científica acepta el modelo Kepleriano que agrupa el heliocentrismo, las leyes de Kepler y las leyes de la mecánica establecidas por Newton (Ponce, 2010).

El problema de los dos cuerpos; es decir, el movimiento de dos masas que interactúan por la aceleración gravitacional puede ser descrita como una función de tiempo y es analíticamente soluble. Gracias a la mecánica clásica Newtoniana es posible describir el movimiento del sistema como una ecuación diferencial por tanto el problema se reduce a calcular las funciones que satisfacen estas ecuaciones (Guzman Perez et al., 2025).

Este problema es de gran importancia en astronomía, mecánica orbital, dinámica galáctica, formación estelar, así como en la determinación de trayectorias óptimas para misiones de naves espaciales (Perezagua López, 2022).

Sin embargo, un sistema mayor a dos cuerpos no es analíticamente soluble por tanto no existe una solución explícita del sistema de ecuaciones. En el presente trabajo, se simulará la trayectoria tanto de dos cuerpos como de múltiples. Mediante la aproximación con el método de Runge Kutta y el uso de redes neuronales físicamente informadas.

Exploración

El ser humano desde el inicio de los tiempos ha presentado cierta fascinación con el espacio. Desde las más antiguas civilizaciones el ser humano ha dedicado tiempo y esfuerzo en entender el movimiento de los cuerpos celestes desde una perspectiva filosófica, espiritual y científica. Primeramente, el interés de estas civilizaciones giraba en torno a las estrellas, lo que probó ser fundamental con el advenimiento de la navegación (Ponce, 2010).

Uno de los mayores quiebres en la teoría se produjo por Ptolomeo en el siglo II d.C, quien propuso que la Tierra ocupaba el centro del universo y los demás cuerpos giraban en órbitas alrededor de ella es decir, un modelo geocéntrico. Claramente, apoyándose con la teoría de Aristóteles; Teoría del geocentrismo. A pesar de no asemejarse a la realidad que conocemos probó ser bastante preciso en sus cálculos a la hora de predecir fenómenos astronómicos (Cortés Maldonado, 2021).

Luego en el siglo XVI, Nicolás Copérnico plantea el Sol como el centro del universo, donde la tierra y demás cuerpos giraban alrededor de él; conocido como la teoría del heliocentrismo. La cual fue luego validada por Galileo usando el recién inventado telescopio (Ponce, 2010). A Galileo se le atribuye como el primero en utilizar el telescopio para el estudio sistemático de los cielos. Asimismo, se le considera “el padre de la ciencia” ya que sus descubrimientos contribuyeron a ponerle fin a la ciega lealtad a la autoridad tanto eclesiástica como a otros pensadores entre ellos Aristóteles, en materia de ciencia. Además, de promulgar la separación de la ciencia, religión y filosofía (Museo Virtual de la Ciencia del CSIC, 2025).

Posteriormente, Johannes Kepler (1571-1630) postula las leyes de Kepler, conocidas como la Ley de las Órbitas, la Ley de las Áreas y la Ley de los Periodos. A partir de estas, Newton llega a ciertas conclusiones importantes. Una de ellas es que los planetas y los satélites no están en equilibrio; hay una fuerza neta que actúa sobre ellos ya que se mueven en órbitas (1era Ley del Movimiento). La segunda es que todos los cuerpos se atraen unos a otros mediante la fuerza gravitatoria, por lo tanto las fuerzas centrales sobre los planetas en el Sistema Solar es la atracción de la fuerza gravitatoria del Sol (Universidad Iberoamericana, 2025).

Una vez que son postuladas y aceptadas las leyes de la mecánica desarrolladas por Newton en 1687, a las cuales también se les conoce como Leyes de Newton, la comunidad científica aprueba la Teoría Heliocéntrica y las las Leyes de Kepler. El modelo resultante de aplicar las leyes de Newton a un sistema con características similares al nuestro es lo que se conoce como el modelo Kepleriano (Ponce, 2010).

La mecánica clásica newtoniana permite describir el movimiento de los objetos mediante ecuaciones diferenciales. Estas ecuaciones basadas en las Leyes de Newton expresan cómo cambian las variables físicas como posición y velocidad en función del tiempo. Al resolver el sistema, se obtiene la trayectoria del mismo en cualquier instante. Por lo tanto, el comportamiento del sistema se reduce en calcular las funciones que satisfacen las ecuaciones. Ahora bien, encontrar las funciones mencionadas es sumamente complejo, inclusive para modelos simples. Los intentos de encontrar estas funciones se agrupan bajo el nombre del problema de los n cuerpos (Ponce, 2010).

Al tratarse de dos cuerpos, sí es posible encontrar una solución al sistema donde se implementa a su vez la fórmula de la fuerza gravitacional para lograrlo. Sin embargo, para un sistema

de tres o más cuerpos esto no es posible por tanto no es “analíticamente soluble” (Guzman Perez et al., 2025). La conclusión de que no es posible encontrar una función en el caso de tres cuerpos fue desarrollada por Henri Poincaré. Él determinó que en casos como el anterior la evolución del sistema es extremadamente caótica. Ya que, una pequeña perturbación en el estado inicial podría llevar eventualmente a un estado totalmente diferente. Por lo tanto, es imposible predecir el estado del sistema dado que los instrumentos de medición ni siquiera pueden detectar esa mínima variación (Pardo Silván, 2022).

Poincaré más bien propone un estudio más cualitativo de las órbitas cuando se trata de esta cantidad o mayor cantidad de masas. Es así, que se establece una nueva área en la matemática; sistemas dinámicos (Ponce, 2010). Asimismo, es posible implementar lo que se conoce como redes neuronales para tratar de predecir las órbitas de los cuerpos. Estas se proponen como un modelo artificial del cerebro humano, cuyo propósito es simular ciertas características como la capacidad de memorizar y asociar hechos. Hay ciertos problemas cuya respuesta no es tan simple como un algoritmo, sino que dependen de la experiencia. Las redes neuronales buscan llenar ese vacío, mediante un sistema de tratamiento de la información basado en la experiencia acumulada (Matich, 2001).

Este modelo de computación prueba ser adecuado para el reconocimiento de patrones con el fin de identificar y clasificar objetos. En este caso, se utiliza para aproximar basado en las velocidades y posiciones de los planetas del sistema solar, las órbitas de tanto estos como otros cuerpos celestes. Estructuralmente, cuenta con una serie de nodos conectados donde se descomponen primeramente las entradas y cuyo comportamiento depende de la forma en que se realizan las conexiones de estos elementos individuales y la importancia de cada conexión. Dichos nodos conforman lo que se conoce como capas las cuales están interconectados tal que cada capa utiliza la salida de la capa anterior como entrada. La estructura más simple consta de una capa de entrada, una capa de oculta y una capa de salida. Sin embargo, las capas ocultas frecuentan ser muchas más (MathWorks, s. f.).

Para entrenar a dichas redes se utilizan técnicas de machine learning entre las que destacan: el aprendizaje supervisado y no supervisado, cuya diferencia radica en si el conjunto de datos se encuentran etiquetados o no, la clasificación quien se entrena para reconocer ciertas categorías, la regresión, el reconocimiento de patrones y el clustering. El uso de estas técnicas depende del problema a abordar, la precisión de los resultados está sujeta a que tanto se adecua esta técnicas a la tarea en mano (MathWorks, s. f.).

Asimismo, a partir del concepto de redes neuronales se desarrolla lo que se conoce como redes neuronales informadas por la física o PINN por sus siglas en inglés, estas redes funcionan como las tradicionales sin embargo son capaces de incorporar leyes de la física. Estos se han llegado inclusive a ser considerados como métodos numéricos ya que primordialmente son empleados en la resolución de problemas con ecuaciones en derivadas parciales especialmente las que su resolución puede llegar a ser altamente compleja mediante métodos numéricos estándar (Ramos Mísea, 2023). Por lo tanto, es ideal ante el problema de los n cuerpos, ya que su formación está basada en estas ecuaciones y su incapacidad de encontrar la solución de las mismas.

Las PINN comúnmente cuentan con una estructura de tres grandes bloques. Primeramente, una red neuronal artificial el cual recibe un vector de variables y devuelve el valor de la solución predicha. Luego, se establece un “módulo físicamente informado” el cual calcula las derivadas

de la salida de la red respecto a sus entradas e introduce las leyes físicas donde determina que tan bien se cumplen estas leyes a los puntos anteriores. Finalmente, cuenta con un mecanismo de retroalimentación que define el problema de minimización. Es decir, considera el error de los datos observados y el error al incluir las ecuaciones físicas como método de ajuste a los parámetros de la red, para posteriormente mejorar su predicción.

Metodología

Basado en lo expuesto en la introducción y la exploración, es pertinente plantear una metodología que permita darle una respuesta satisfactoria al problema planteado. En este caso, se pretende lograr tres objetivos principales: aproximación, simulación y animación. Como se explicó en la sección anterior, el problema de los dos cuerpos puede ser resuelto por completo de forma analítica, pero esto requeriría darle un enfoque al trabajo que no solo excede los alcances del curso, sino que se desvía de los aprendizajes esperados.

Por esta razón, para el problema de los dos cuerpos, la implementación se limitará a realizar una aproximación de la órbita, preferiblemente en un sistema bien conocido como Tierra-Sol, o de manera más general, de un cuerpo cuya masa pueda expresarse en términos del otro objeto en el sistema. Para abordar esta tarea, se puede crear una clase dedicada que, basada en los procedimientos propuestos por Alhowaity (2022), contenga un método que permita aproximar la solución al problema de los dos cuerpos, tomando como parámetros: tiempo inicial, tiempo final y los vectores de posición y velocidad iniciales. No obstante, para obtener estos valores es necesario realizar varios pasos previos, como el cálculo de los coeficientes requeridos para las series de potencias que se definen en el artículo previamente citado. Como resultado final de los métodos de esta clase, se espera obtener una tabla con los valores en cada coordenada (x,y,z) de los vectores de posición y velocidad del objeto a lo largo de un lapso establecido. Cabe destacar que un beneficio de realizar una aproximación y no buscar una solución exacta, es que queda margen para comparar los datos generados por el modelo con cifras de bases de datos reales, para así evaluar su porcentaje de error; esto será explorado en la implementación.

El segundo aspecto que se espera cubrir dentro de la implementación es la simulación. El primer paso debe ser la creación de una clase “Cuerpo”, que pueda representar un planeta, una estrella o cualquier otro objeto astronómico significativo, por lo que sus atributos deben necesariamente incluir su posición inicial (en tres dimensiones), su velocidad y su masa. A partir de esta primera clase madre, se puede desarrollar otra clase que se encargue de la simulación propiamente, para lo cual debería tener como atributo la lista de cuerpos (creados con la clase anterior) que van a interactuar y el tiempo que va a transcurrir en la simulación.

Ahora, para obtener los valores de posición y velocidad de los cuerpos a través del tiempo, se propone utilizar el Método de Runge-Kutta de cuarto orden (RK4), un método iterativo para la aproximación de soluciones a ecuaciones diferenciales. Este es relevante porque en el problema de los n -cuerpos, la posición y velocidad de cada cuerpo cambian con el tiempo por la influencia de las fuerzas gravitacionales de los demás, lo cual puede describirse mediante ecuaciones diferenciales. No solo ofrece soluciones para cada intervalo de tiempo establecido, sino que además utiliza un promedio ponderado de cuatro estimaciones de la pendiente de

la solución para mejorar la precisión. Por lo tanto, es indispensable que la clase “Simulación” incorpore un método capaz de ejecutar el RK4 considerando varios cuerpos, para lo cual también es necesario un método auxiliar que calcule las aceleraciones y otro que actualice las posiciones de los cuerpos tras cada estimación.

Finalmente, se espera incluir una funcionalidad de animación dentro del código. Hasta el momento, los módulos anteriores solo muestran resultados numéricos o almacenan las operaciones internamente, pero para representar más intuitivamente su comportamiento se considera oportuno agregar módulos con la capacidad de mostrar gráficamente las trayectorias de los cuerpos en dos y tres dimensiones. Para esto, se incluirá una clase “Simulación2D” que herede de “Simulación”, la cual obtendrá los datos de las posiciones en el tiempo para dibujar las trayectorias con el uso de paquetes de graficación y animación como `matplotlib.pyplot` y `matplotlib.animation`. El principal método tendrá que definir de manera dinámica el rango de los ejes para garantizar la correcta visualización de todos los cuerpos, crear puntos que representen los cuerpos, líneas para trazar las trayectorias y actualizar periódicamente el gráfico mostrado para cubrir la órbita completa (o al menos en el periodo que se defina). De forma similar para la visualización en tres dimensiones, se construirá una clase “Simulación3D” con un funcionamiento análogo a su contraparte bidimensional, solo que con una coordenada adicional.

Implementación

Como se mencionó en la sección de Metodología, para el problema de los dos cuerpos se decidió aproximar la solución de la órbita y a la vez simularla. Aunque la solución de este problema existe tiene muchas variantes según cuales suposiciones iniciales se tomen, entre ellas la conservación de energía, de momento angular, deformaciones en espacio-tiempo, tiempo en actuar la fuerza gravitacional, entre otros (Helm, 2022).

Para la aproximación de los dos cuerpos, se tomó como guía el artículo publicado por Alhwaity (2022) donde utiliza series de potencias para aproximar la solución de las ecuaciones diferenciales. Se sigue el mismo método propuesto en el artículo con los 10 coeficientes de la suma debido a que obtener los siguientes resulta complejo y computacionalmente más demandante. Así en la clase `SeriesAproximar.py` se obtienen los coeficientes y las aproximaciones para una cantidad arbitraria de días, además como se mencionó en la metodología se usarán rangos pequeños de días por el uso de las ecuaciones diferenciales con delays o retrasos.

Con el polinomio ya calculado se crea un data frame en el que para cada día en el rango dado se genera una observación con un valor calculado de posición en las tres coordenadas y velocidad en las tres coordenadas. Con esto en el script de R denominado como `LimpiezaDatos.R` se carga la base de datos descargada en la página de la NASA denominada [Jet Propulsion Laboratory](#) donde se obtuvieron los datos de posición y velocidad para la Tierra tomando como observador el Sol, comenzando el día 01-04-2025 y terminando el 05-06-2025. El módulo de R provee un breve código para leer la tabla y darle formato a la columna de fecha, además de seleccionar las columnas importantes. Es ahí donde en el script `AnalisisDatos.R` se toma esta base llamando el script `LimpiezaDatos.R` y la base de datos calculada llamando el script `main.py`, uniendo

ambas tablas, calculando la norma de los vectores de posición y velocidad y generando los porcentajes de error.

Fecha	Posicion (UA)			Velocidad (UA/dia)		
	Real	Calculada	Error	Real	Calculada	Error
2025-04-01	0.99922	0.93917	6.01%	0.01722	0.35912	1985.25%
2025-04-02	0.99951	0.93917	6.04%	0.01722	0.35912	1986.00%
2025-04-03	0.99979	0.93917	6.06%	0.01721	0.35912	1986.76%
2025-04-04	1.00007	0.93917	6.09%	0.01720	0.35912	1987.55%
2025-04-05	1.00035	0.93917	6.12%	0.01720	0.35912	1988.34%

Elaboración propia con datos reales de: NASA/JPL Horizons System

Tabla 1: Comparacion entre valores calculados y reales: Posicion y velocidad de la Tierra con respecto al Sol

En la anterior tabla se puede apreciar un pequeño análisis en el porcentaje de error tanto de la posición del cuerpo como de la velocidad, los detalles de este serán mencionados en el FODA.

Posteriormente con la clase `Cuerpo.py` mencionada en la metodología se instanciaron las características del objeto astronómico, posición inicial, velocidad inicial y masa, note que tanto el vector de posición como de velocidad acepta tanto de 2 o 3 coordenadas. Una vez instanciados los objetos tipo `Cuerpo`, la clase `Simulacion_N_Cuerpos.py` recibe una lista de ellos para los cuales calcula sus trayectorias aplicando el método de Runge Kutta de orden 4, esto lo hace con un tiempo de paso definido y recalculando su aceleración y posición con los métodos auxiliares programados en la misma clase. Con esto, el último método llamado `simular` genera un bucle con el tiempo de pasos definidos para crear un ‘gif’ con la simulación. En este apartado, es importante considerar que si se coloca un parámetro ‘h’ muy alto puede que en la simulación no se aprecie, o la aceleración y trayectorias no sean las esperadas. Esto se debe a que con un paso muy grande en el método de Runge Kutta el cálculo del nuevo punto por el que pasa el cuerpo será muy lejos del anterior punto demarcado, haciendo que el factor de la distancia entre cuerpos sea insignificante en el cálculo.

La clase `Simulacion_2D.py` propone un método que junto con los módulos mencionados en la metodología (`matplotlib.pyplot` y `matplotlib.animate`) es capaz de crear ejes dinámicos, que se ajustan según las posiciones iniciales de los cuerpos y una animación donde se aprecie el movimiento de los planetas. A su vez se tiene un método para guardar la animación como un gif.

La clase `Simulacion_3D.py` propone un método similar, solo que considerando un plano con 3 dimensiones, pero siguiendo la misma dinámica que `Simulacion_2D`.

Análisis FODA

En la siguiente tabla se comentan las fortalezas, oportunidades, debilidades y amenazas identificadas en el desarrollo actual del proyecto:

Fortalezas	Oportunidades	Debilidades	Amenazas
Métodos y procedimientos con amplio respaldo bibliográfico y ejemplos pasados que demuestran su funcionamiento.	En el módulo SeriesAproximar, existe la posibilidad de agregar polinomios adicionales para mejorar la precisión del modelo en periodos de tiempo más extensos, pero la viabilidad de esto debe analizarse con mayor profundidad más adelante.	Con los métodos actuales, el porcentaje de error en los vectores de posición para un sistema Tierra-Sol está entre 6% y 6,4%, que es considerable.	Al ser un sistema caótico, obtener datos reales con suficiente precisión para comparar simulaciones de más de dos cuerpos es complejo. lo que puede limitar la evaluación objetiva del modelo.
En general, se cumplen los objetivos planteados en la metodología: simulación y animación de sistemas de n-cuerpos, así como aproximación para un escenario de dos cuerpos.	Se puede investigar y experimentar sobre técnicas para reducir el porcentaje de error de los resultados obtenidos con el modelo respecto a datos reales.	Los porcentajes de error en los vectores de velocidad son aún mayores y, al menos por el momento, se desconoce cómo mejorarlos.	El modelo es más simple y menos preciso que otras alternativas profesionales como REBOUND, que puede admitir muchos más parámetros iniciales, pero al costo de mayor exigencia computacional.

Fortalezas	Oportunidades	Debilidades	Amenazas
<p>Código bien estructurado, dividido en módulos con propósitos claros y que interactúan correctamente entre sí.</p> <p>Se logró comparar los datos obtenidos mediante el modelo con datos reales, lo que permitió conocer su margen de error.</p>		<p>El módulo SeriesAproximar debe usarse con periodos cortos, ya que al depender de ecuaciones diferenciales con retraso, la aproximación para intervalos de tiempo más grandes puede resultar en márgenes de error considerables.</p>	

Anexos

Repositorio de GitHub : https://github.com/DebbieCon/simulacion_planetas_CA0305/blob/main/simulacion_planetas_CA0305.qmd

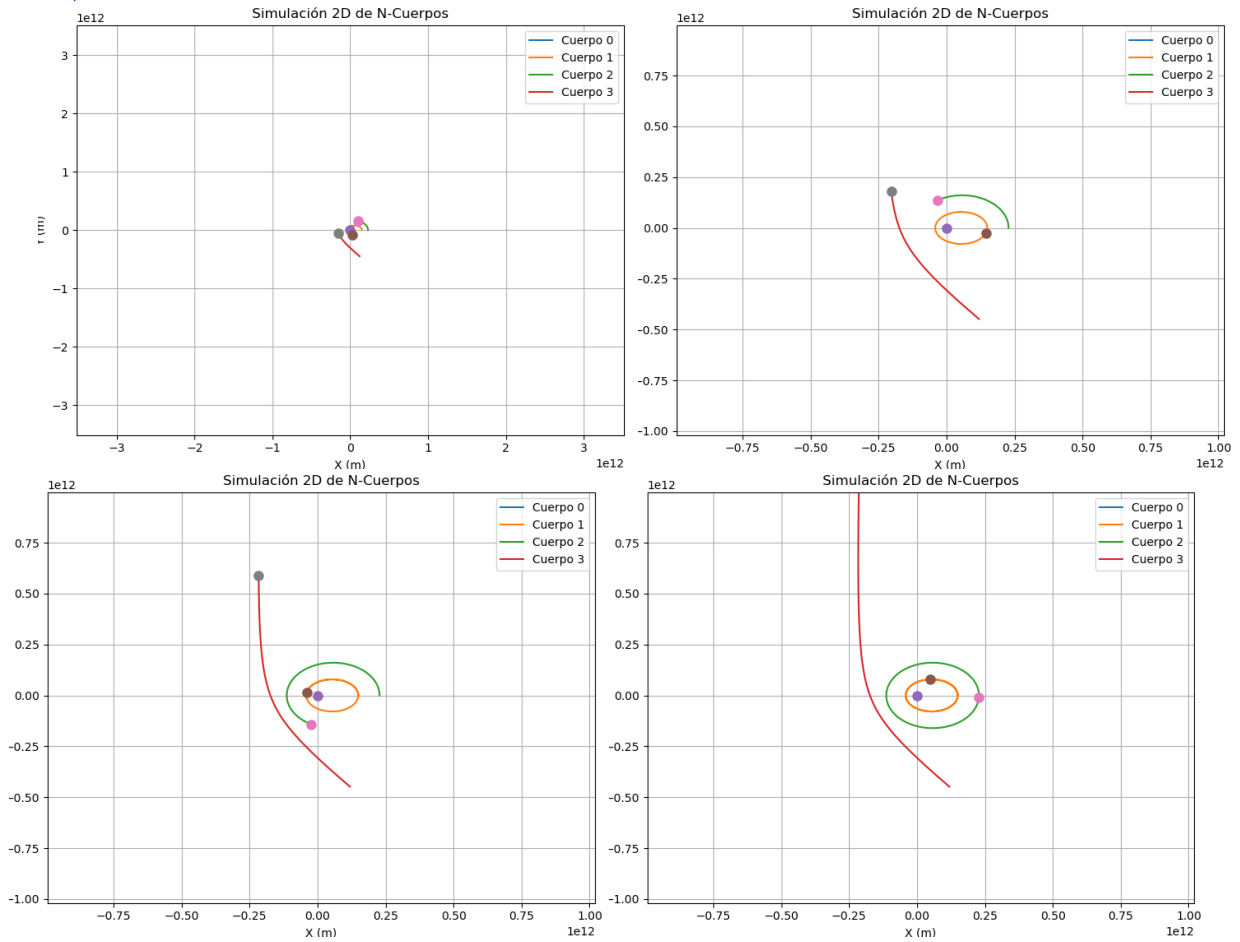


Figura 1: Simulación 2D de un sistema conformado por una estrella quieta masiva, 2 planetas y un cometa.

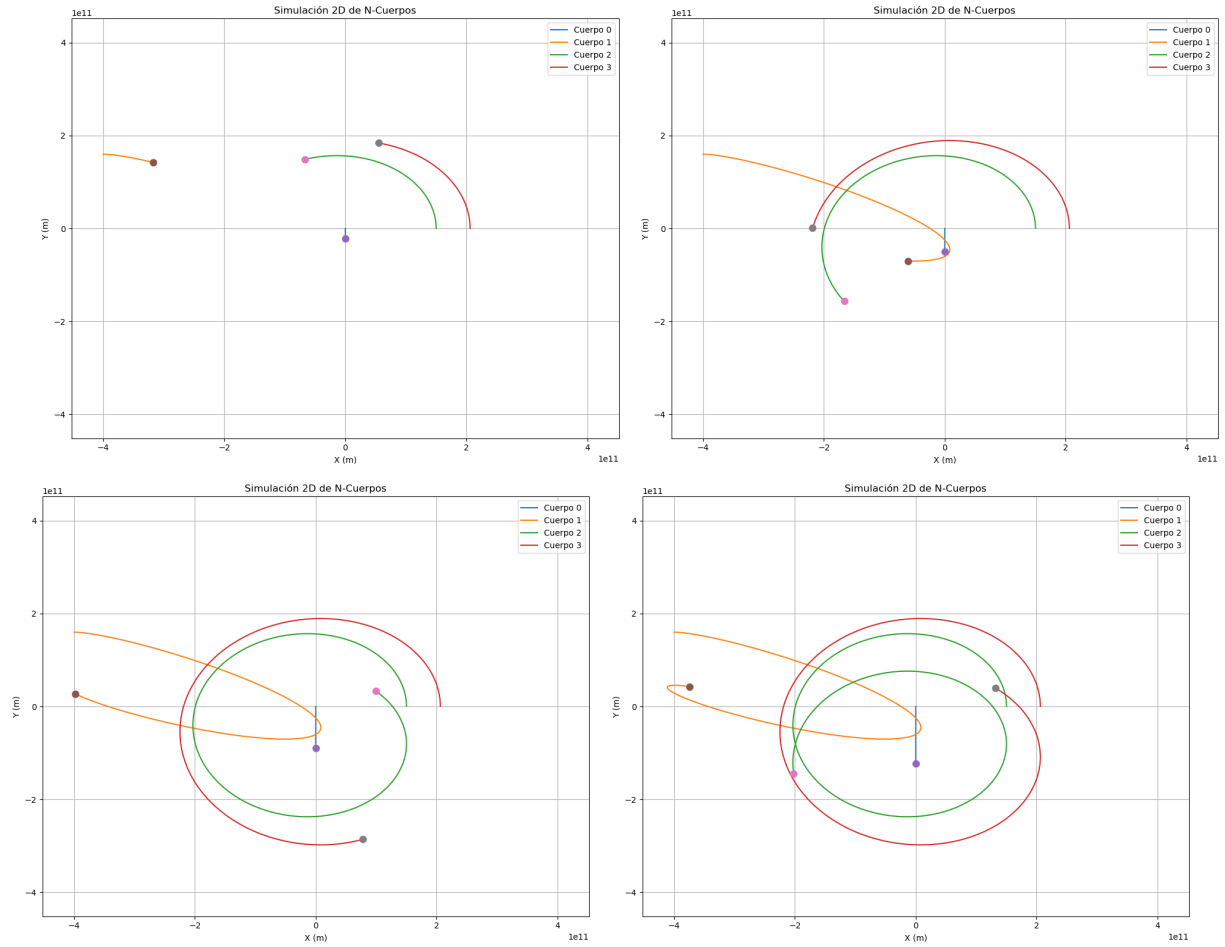
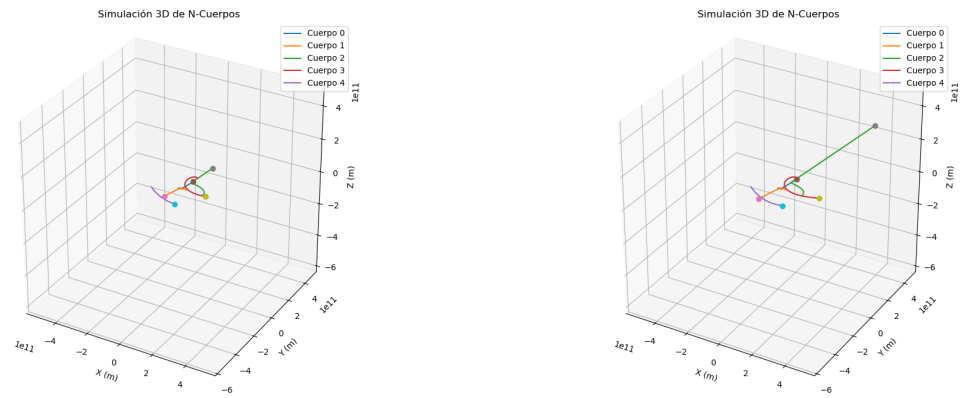


Figura 2: Simulación 2D de un sistema conformado por el Sol, Tierra, Marte y un cuerpo intruso.



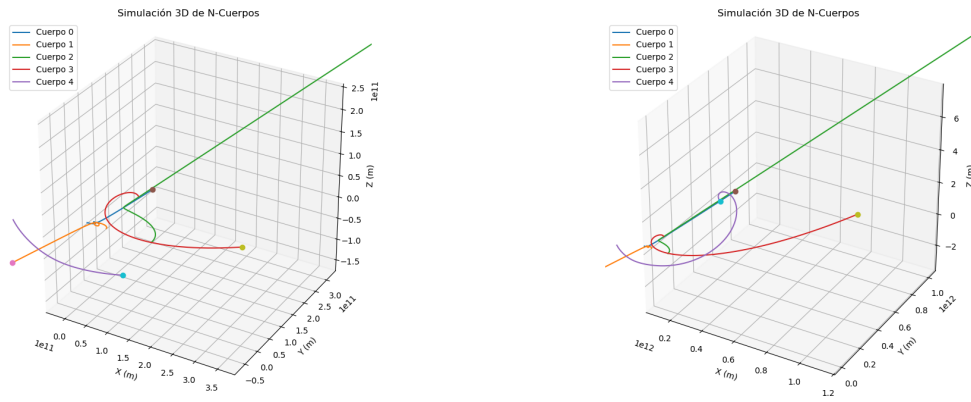


Figura 3: Simulación 3D de un sistema conformado por una estrella quieta, dos planetas y dos cometas.

Referencias

- Alhawaity, S. (2022). Computational algorithm to solve two-body problem using power series in geocentric system. *Applied Mathematics and Nonlinear Sciences*, 1-8. <https://doi.org/10.2478/amns.2021.2.00300>
- Cortés Maldonado, L. Y. (2021). *Aproximación al problema de los dos cuerpos* [Mathesis, Universidad Pedagógica Nacional; <https://repositorio.pedagogica.edu.co/bitstream/handle/20.500.12209/13457/APROXIMACION%20AL%20PROBLEMA%20DE%20LOS%20DOS%20CUERPOS.pdf?sequence=1>]. <https://repositorio.pedagogica.edu.co/handle/20.500.12209/13457>
- Guzman Perez, Y. U., Fragoso Martínez, I., & García Mendoza, F. (2025). *El problema de los dos cuerpos*. Proyecto de Análisis Numérico III, Escuela Superior de Física y Matemáticas, Instituto Politécnico Nacional.
- Helm, J. (2022). Approximate and Exact GR-Solutions for the Two-Body Problem. *Journal of High Energy Physics, Gravitation and Cosmology*, 8(3), 690-723. <https://doi.org/10.4236/jhepgc.2022.83051>
- MathWorks. (s. f.). ¿Qué es una red neuronal? En *¿Qué es una red neuronal? - MATLAB & Simulink*. <https://la.mathworks.com/discovery/neural-network.html>
- Matich, D. J. (2001). *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Universidad Tecnológica Nacional – Facultad Regional Rosario.
- Museo Virtual de la Ciencia del CSIC. (2025). *Galileo Galilei*. <https://museovirtual.csic.es/salas/magnetismo/biografias/galileo.htm>
- Pardo Silván, E. (2022). *Mecánica Hamiltoniana y el problema de los n cuerpos* [Trabajo Fin de Grado]. Universidad Politécnica de Madrid. <https://oa.upm.es/71786/>
- Perezagua López, R. L. (2022). *El problema de los tres cuerpos*. Empresarios Agrupados. <https://www.empresariosagrupados.es/wp-content/uploads/2022/12/Art.-3-cuerpos.pdf>

- Ponce, M. (2010). *Movimiento de planetas y otras dinámicas de baja dimensión*. Facultad de Matemáticas, PUC-Chile; Notas del mini curso dictado en la XXXVI Semana de la Matemática.
- Ramos Misa, J. M. (2023). *Introducción a las Redes Neuronales Físicamente Informadas*. <http://hdl.handle.net/10498/30551>
- Universidad Iberoamericana. (2025). *5. Leyes de Kepler*. Universidad Iberoamericana; <https://ibero.mx/campus/publicaciones/fisica/pdf/5kepler.pdf>.