

# Simulación de N-Cuerpos con el algoritmo de Barnes-Hut

Laura Vivina Alfonso Díaz

*Departamento de física, Universidad Nacional de Colombia*

Daniel Felipe Rojas Paternina

*Departamento de física, Universidad Nacional de Colombia*

David Calderón Gómez

*Departamento de física, Universidad Nacional de Colombia*

Es bien conocido que el problema de dos cuerpos interactuando mediante un potencial central tiene una solución analítica. Ya el problema de tres cuerpos no corre con la misma suerte, y en general el problema de muchos cuerpos no tiene una solución analítica. El planteamiento de la dinámica de un sistema de  $N$  cuerpos no representa un problema teórico, por cuanto se trata simplemente de obtener la ecuación de movimiento para cada uno de los cuerpos. Considere que todos los cuerpos interactúan mutuamente, exceptuando la interacción propia. En ese sentido, la ecuación de movimiento para el cuerpo  $i$  queda como:

$$\frac{d^2 \vec{x}_i}{dt^2} = \sum_{j=1, j \neq i}^N \frac{Gm_j (\vec{x}_j - \vec{x}_i)}{||\vec{x}_j - \vec{x}_i||^3} \quad (1)$$

Se trata nada más que de la segunda ley de Newton. Cada una de las ecuaciones depende de las variables dinámicas de todas las partículas. Se trata de un sistema de  $N$  ecuaciones diferenciales acopladas de segundo orden, que constituye un problema computacional formidable. Existen muchos algoritmos dedicados a solucionar las ecuaciones de movimiento de la mecánica clásica, llamados algoritmos simplécticos. (Un algoritmo se dice simpléctico si preserva el volumen en el espacio de fase, es decir, satisface el teorema de Liouville).

## Algoritmo de Barnes Hut

Acometer el problema inmediatamente pueden representar problemas técnicos. Para un sistema de  $N$  cuerpos, el número de cálculos que se deben hacer crece muy rápidamente. Considere el caso de tres cuerpos, dos de ellos muy cercanos entre sí. La ecuación de movimiento para cada uno de los cuerpos, queda como:

$$\frac{d^2 \vec{x}_1}{dt^2} = \frac{Gm_2 (\vec{x}_2 - \vec{x}_1)}{||\vec{x}_2 - \vec{x}_1||^3} + \frac{Gm_3 (\vec{x}_2 - \vec{x}_1)}{||\vec{x}_2 - \vec{x}_1||^3} \quad (2)$$

Dado el caso que dos de los tres cuerpos se encuentren muy cercanos, las fuerzas de interacción con el tercer cuerpo son muy similares, ya que sólo depende de la distancia y de la masa. Una buena simplificación sería considerar que los dos cuerpos cercanos constituyen uno único con masa total igual a la suma de las masas y posición igual al centro de masa. Podemos obtener una solución aproximada del mismo sistema, aun cuando hemos reducido el número de contribuciones. Esta es la idea central del algoritmo de Barnes Hut. No se trata de un algoritmo para resolver las ecuaciones

diferenciales, sino para reducir el número de fuerzas significantes para la descripción del sistema.

Considere un cuerpo arbitrario de un sistema de  $N$  cuerpos. Las fuerzas que actúan sobre este cuerpo son  $N - 1$ . Considere ahora un cúmulo de  $k$  masas suficientemente alejadas como para ser consideradas como una única masa, situada exactamente en el centro de masa. Enés de calcular estas  $k$  fuerzas, calcularemos sólo una, esto reduce el número de fuerzas calculadas en  $k - 1$ . El proceso se adelanta para cada una de las masas en cada uno de los pasos de tiempo, de manera sistemática.

Considere el sistema de cuerpos en dos dimensiones, contenidos en un cuadrado. Divida este cuadrado en cuatro cuadrados más pequeños. Si alguno de estos cuadrados queda una única masa, o ninguna, este cuadrado no se dividirá más. De lo contrario se dividen los cuadrados más pequeños nuevamente en cuatro. Así se adelanta el proceso sucesivamente hasta que cada cuadrado contenga una o ninguna masa. Organizaremos así el sistema en una estructura de árbol. Identificaremos cada cuadrado con un nodo. Los nodos sin masa se dicen nodos vacíos. Los nodos con una única masa si dicen hojas. El nodo principal se dice raíz, y corresponde con el área total en la que se encuentra el sistema, y los nodos con más de una masa se dicen nodos intermedios.

Todas las hojas guardan la información de la masa y la posición del cuerpo. Los nodos intermedios guardan la información de la masa total y del centro de masa de todos los cuerpos contenidos en él. El algoritmo de Barnes Hut es simple de entender, pero la estructura recursiva del árbol y su construcción es verdaderamente fundamental. De esta forma, las fuerzas pueden ser calculadas recorriendo los nodos del árbol, lo que permite acelerar significativamente la simulación. Este proceso de recorrido implica la aplicación de un criterio, el cual determina cuándo un grupo de cuerpos es suficientemente lejano para ser tratado como una sola partícula, y así no se requiere calcular todas las interacciones entre las partículas. A continuación se muestra la estructura del árbol:

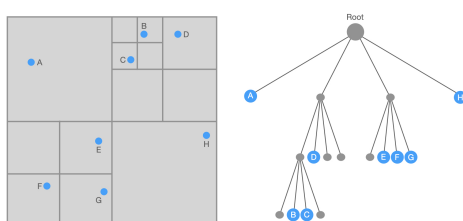


Figure 1: Árbol de regiones espaciales en el algoritmo de Barnes-Hut.

Suponga que quiere calcular las fuerzas que actúan sobre una masa. Y considere respecto a este cuerpo un nodo intermedio. Si la razón entre la distancia al centro del nodo y la distancia al centro de masa es menor que cierto parámetro,  $\theta$  se admite la simplificación de Barnes Hut, de lo contrario se procede a evaluar el mismo argumento sobre los nodos hijos. Este procedimiento se adelanta exhaustivamente hasta llegar a las hojas. El algoritmo de Barnes Hut se puede emplear en una dimensión, en el que el espacio se divide secuencialmente por la mitad, o en tres dimensiones dividiendo el espacio secuencialmente en octantes.

## Método de integración de Verlet

Se trata de un método numérico usado para resolver las ecuaciones de movimiento de Newton. Se trata de un algoritmo simpléctico y que además tiene simetría respecto a la inversión de tiempo. Este algoritmo se popularizó en primer lugar al ser usado en simulaciones de dinámica molecular. La posición de un cuerpo dado un paso anterior está dada por:

$$\vec{x}_{n+1} = 2\vec{x}_n - \vec{x}_{n-1} + \vec{a}_n \Delta t^2 \quad (3)$$

El algoritmo es más estable que el de Euler porque considera información pasada. Este algoritmo básico sólo nos permite conocer las posiciones. La velocidad podría calcularse usualmente como el valor medio, sin embargo, existe una variante de Verlet para calcular las velocidades, similar al método *leapfrog*. La velocidad se da como:

$$\vec{v}_{n+1} = \vec{v}_n + \frac{1}{2}(\vec{a}_n + \vec{a}_{n+1})\Delta t \quad (4)$$

Este algoritmo es de orden 2, existen algoritmos mucho más precisos de orden 4 como el PEFRL. Sin embargo, para los propósitos presentes, el algoritmo de Verlet presenta resultados aceptables.

## Unidades del modelo

Nuestra intención fue simular el movimiento de las estrellas en una galaxia espiral, en la cual hay una gran variedad de cuerpos con distintas masas y dimensiones, estos objetos poseen masas del orden de la masa solar, por esto hemos escogido las masas solares  $M_\odot$  como unidad apropiada para nuestra simulación. Así, se generaron 500 estrellas de forma aleatoria con masas de  $1M_\odot$  a  $50M_\odot$ . Por otra parte, una de las características más esenciales para las galaxias espiral es el agujero negro en su centro, en nuestro caso, escogimos el agujero negro Sagitario A\* que se encuentra en el centro de la Vía Láctea, con una masa de  $3,7 \pm 0,2$  millones de veces la masa solar la cual aproximamos a  $4 \times 10^6 M_\odot$ .

Por otra parte, teniendo en cuenta que las distancias entre estrellas es muy grande, se decidió escoger como unidad de distancias el kilo parsec ( $kpc$ ), el cual se define como, 3262 años luz y es usado frecuentemente para medir largas distancias de objetos astronómicos fuera del sistema solar. De manera que nuestra galaxia simulada tendrá  $30kpc$  de radio. Además, como el movimiento de las estrellas es notable para unidades temporales grandes, se escogió la unidad de años ( $yr$ ) para realizar la simulación y se hizo evolucionar el sistema de 500 cuerpos por  $2000yr$  con un paso de  $5 \times 10^{-3} yr$  para el algoritmo de Verlet.

Por último, es necesario que las velocidades iniciales de las estrellas sean velocidades orbitales, con el fin de que se acomoden de manera apropiada al movimiento de los cuerpos celestes, por este motivo se escogió una velocidad inicial Kepleriana, es decir,

$$\Omega_k(r) = \sqrt{\frac{GM}{r^3}} \quad (5)$$

con estas características, se utilizó el método de Barnes-Hut mencionado anteriormente.

## Resultados

El principal resultado de nuestra simulación, es un gif en el que se evidencia el movimiento de 500 estrellas al rededor de un agujero negro, para el cual cuál se tiene que la interacción gravitacional teóricamente preserva la energía y el momento angular, ya que el potencial tiene simetría esférica y

utilizamos un algoritmo simpléctico. En la gráfica 2 se puede evidenciar el comportamiento de las energías del sistema y en la figura 3 las variaciones al respecto. Como se nota en la escala vertical, estas diferencias son varios órdenes de magnitud menores que la energía total, por lo que podemos asumir que las energías se conservan y se está evidenciando variaciones debido a los supuestos de nuestro modelo y al error computacional de la máquina.

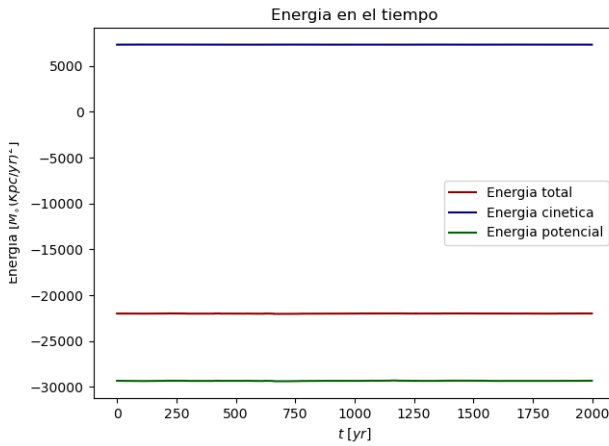


Figure 2: Energía en el tiempo

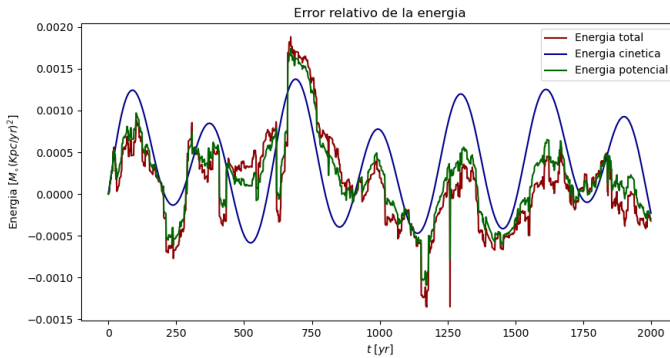


Figure 3: Error relativo de la energía

En la gráfica 4 se muestra la diferencia relativa para el momento angular, de nuevo las diferencias son de un orden de magnitud  $-6$ , insignificantes frente al momento angular total que es de al rededor de  $217000 M_{\odot} kpc/year$  y por lo cual podemos asumir nuevamente que se conserva el momento angular como debe suceder en la simulación.

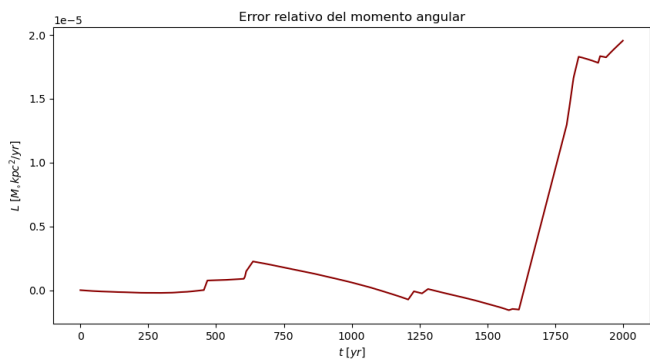


Figure 4: Error relativo del momento angular

Por último, en la figura 5 se muestra la velocidad angular contra a la distancia radial de los cuerpos. Aquí podemos evidenciar que velocidad decae como el inverso de la raíz aproximadamente, por lo que las estrellas más cercanas al agujero negro tienen mayor velocidad que las estrellas distantes, lo cual es una característica de sistemas galácticos clásicos y concuerda con lo evidenciado en la simulación.

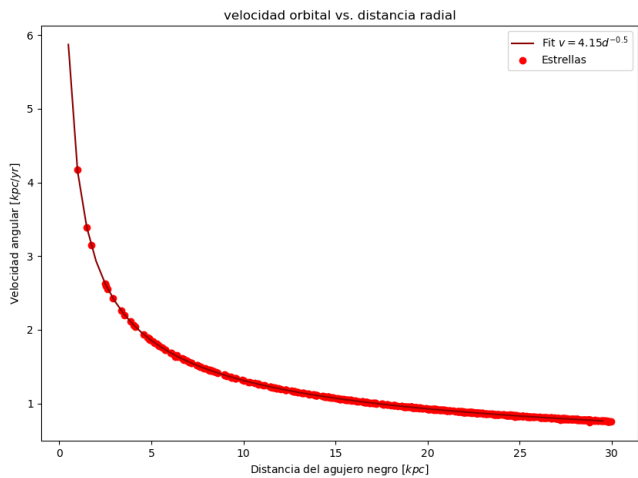


Figure 5: Velocidad orbital vs la distancia radial

## Referencias

1. Swinehart, C. (s.f.). The Barnes-Hut Algorithm. <http://arborjs.org/docs/barnes-hut>.
2. Aguirre Pascual de Zulueta, Maia. (2020-12-15). Estudio e implementación del algoritmo Barnes-Hut para el cálculo de la interacción gravitatoria entre N-cuerpos. <http://hdl.handle.net/10810/49069>.
3. Singh J.P., Holt C., Totsuka T., Gupta A., Hennessy J. (June 1995). Load Balancing and Data Locality in Adaptive Hierarchical N-Body Methods: Barnes-Hut, Fast Multipole, and Radiosity. <https://doi.org/10.1006/jpdc.1995.1077>.
4. Sofue, Y. (2016-12-5). Rotation and mass in the Milky Way and spiral galaxies. Publications of the Astronomical Society of Japan, 69(1), R1. <https://doi.org/10.1093/pasj/psw103>.
4. Sofue, Y. (2016-12-5). Rotation and mass in the Milky Way and spiral galaxies. Publications of the Astronomical Society of Japan, 69(1), R1. <https://doi.org/10.1093/pasj/psw103>.
5. Young, P. (2013) The leapfrog method and other “Symplectic” algorithms for integrating Newton’s laws of motion. <http://physics.ucsc.edu/peter/242/leapfrog.pdf> (Accessed: 23 June 2023).