

**UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE FÍSICA DE SÃO CARLOS**

**Marcos Vinicius Tomás Olegario**

**Desenvolvimento de simulação de n-corpos para o estudo  
da matéria escura**

**São Carlos**

**2024**



**Marcos Vinicius Tomás Olegario**

**Desenvolvimento de simulação de n-corpos para o estudo  
da matéria escura**

Trabalho de conclusão de curso apresentado  
ao Programa de Graduação em Física  
do Instituto de Física de São Carlos,  
da Universidade de São Paulo, para a  
obtenção do título de Bacharel em Física  
Computacional.

Orientador: Prof. Dr. Luiz Vitor de Souza  
Filho

**São Carlos**  
**2024**

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

olegario, Marcos Vinicius Tomás  
Desenvolvimento de simulação de n-corpos para o estudo  
da matéria escura / Marcos Vinicius Tomás Olegario;  
orientador Luiz Vitor de Souza Filho -- São Carlos, 2024.  
27 p.

Trabalho de Conclusão de Curso (Bacharel em Física  
Computacional) -- Instituto de Física de São Carlos,  
Universidade de São Paulo, 2024.

1. Problema de N-Corpos. 2. Matéria Escura. 3. Gadget-  
2. 4. Simulação numérica. I. Souza Filho, Luiz Vitor de,  
orient. II. Título.

## RESUMO

Neste trabalho, discutimos as simulações de n-corpos, uma importante ferramenta para o estudo de uma das principais componentes do Universo: a matéria escura. A partir da implementação de algoritmos para a resolução numérica dos problemas de dois e três corpos, descrevemos as suas características e um código para a análise de sistemas com um número qualquer de corpos que interagem unicamente através da gravidade. Apresentamos, também, um exemplo de simulação com condições iniciais que descrevem a colisão de duas galáxias espirais e realizamos uma comparação da eficiência do nosso programa com o código **GADGET-2**, utilizado para simulações cosmológicas de n-corpos. Com isso, concluímos a ineficiência de nosso programa para tratar um número de corpos maior que  $10^3$ . Entretanto, a sua precisão se manteve apropriada, com erro relativo para a energia mecânica estimado na ordem de  $10^{-3}$  quando testado com 60 mil corpos. Complementarmente, utilizamos o poder computacional do supercluster *Santos Dumont*, do LNCC, para complementar as análises de nossas simulações.

**Palavras-chave:** Problema de N-Corpos. Matéria Escura. Gadget-2. Simulação Numérica.



## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>7</b>
<b>2</b>	<b>DESENVOLVIMENTO</b>	<b>9</b>
<b>2.1</b>	<b>Problema de Dois Corpos</b>	<b>9</b>
2.1.1	Solução Analítica	10
2.1.2	Solução Numérica	12
<b>2.2</b>	<b>Problema de Três Corpos</b>	<b>15</b>
2.2.1	Problema de Três Corpos Restrito	16
<b>2.3</b>	<b>Simulação de n-corpos</b>	<b>18</b>
2.3.1	Arquitetura da Simulação Implementada	19
2.3.2	Suavização Gravitacional	20
<b>2.4</b>	<b>O Código GADGET-2</b>	<b>20</b>
2.4.1	Exemplo de Execução: colisão de galáxias espirais	21
<b>3</b>	<b>RESULTADOS</b>	<b>24</b>
<b>4</b>	<b>CONCLUSÃO</b>	<b>26</b>
	<b>REFERÊNCIAS</b>	<b>27</b>



# 1 Introdução

Um resultado amplamente aceito na comunidade científica atualmente é de que o principal constituinte da parte massiva de nosso Universo é um componente desconhecido chamado de Matéria Escura (DM, do inglês Dark Matter). A sua presença corresponde a 23% (1) de todo o cosmos e a origem do termo está associada à ausência de interações eletromagnéticas da DM com o conteúdo luminoso do Universo, denominado matéria bariônica. Desde o século XX, diversas evidências da existência da DM foram acumuladas, e sua presença pode ser inferida a partir do comportamento das curvas de rotação de galáxias, efeitos de lentes gravitacionais e anisotropias na radiação cósmica de fundo.

As bases históricas do paradigma da matéria escura remontam de meados da primeira metade do século XX, quando os astrônomos Ernst Öpik e Jan Hendrik Oort (em 1922 e 1932, respectivamente (1)) relataram que o conteúdo luminoso presente em algumas galáxias não era massivo o suficiente para explicar a sua dinâmica de maneira satisfatória. Em 1933 o conceito de matéria escura foi adotado, quando o astrônomo suíço Fritz Zwicky cunhou o termo após estudar o aglomerado de galáxias Coma e inferiu, a partir do teorema do Virial, que a massa do cluster deveria ser 400 vezes maior que a massa luminosa observada para que o sistema estivesse em equilíbrio dinâmico (2).

Porém, somente em 1970 a hipótese da DM foi consolidada, a partir do trabalho dos astrônomos Vera Cooper Rubin e William Kent Ford sobre curvas de rotação de galáxias espirais (1). A partir de medidas de velocidade mais precisas que as existentes até então, eles observaram que existia uma lacuna na massa esperada para as galáxias, de modo que a quantidade de matéria luminosa não era o suficiente para explicar a rotação das estrelas distantes do centro. Tal resultado poderia ser explicado com uma alteração no perfil de densidade das galáxias, adicionando-se um halo de matéria não visível.

Dada a ausência de interações eletromagnéticas, a gravidade é a principal interação sofrida pela matéria escura. Em virtude disso, as simulações numéricas de n-corpos são uma importante ferramenta no estudo da distribuição de DM em sistemas físicos, uma vez que podemos acompanhar a evolução das posições e velocidades das partículas de DM durante a evolução do sistema sem precisar de muitas suposições acerca da composição delas. Os físicos Navarro, Frenk e White contribuíram de maneira notória para esse estudo. A partir do trabalho sobre a estrutura de halos de matéria escura não relativística (3), eles forneceram, com o uso de simulações de n-corpos, um modelo geral para a distribuição de DM em halos galácticos. Essa parametrização, conhecida como perfil NFW, permitiu estimar de maneira quantitativa a massa luminosa faltante nas galáxias somente a partir dos dados observacionais de curvas de rotação.

Nosso objetivo neste trabalho é estudar os conceitos fundamentais que estão associados às simulações de n-corpos, implementando um código em *python* que nos permite

acompanhar a evolução de sistemas com um número qualquer de partículas que interagem unicamente através da gravidade e comparando o desempenho de nosso programa com um código já existente na literatura da área, o **GADGET-2**.

## 2 Desenvolvimento

### 2.1 Problema de Dois Corpos

A teoria por trás das simulações de n-corpos baseia-se, principalmente, na resolução do chamado problema de n-corpos. Dado um número de partículas interagindo entre si, o problema consiste em determinar as trajetórias em função do tempo para cada corpo. O primeiro a explorar matematicamente esse problema foi Isaac Newton (4) em sua obra *Princípios Matemáticos da Filosofia Natural*, quando resolveu o problema de forma analítica para dois corpos interagindo gravitacionalmente. Newton baseou a sua solução puramente em métodos geométricos, nesta seção, apresentamos uma outra abordagem, utilizada em cursos de graduação, para a solução do problema de dois corpos: a partir do cálculo diferencial e integral. Com isso, temos o objetivo de introduzir o problema para, em seguida, expandi-lo para um número maior de corpos.

Inicialmente, consideramos em um referencial inercial dois corpos pontuais de massas  $m_1$  e  $m_2$  com vetores posição  $\vec{r}_1$  e  $\vec{r}_2$  com relação a origem do sistema, tal qual mostrado na Figura 1.

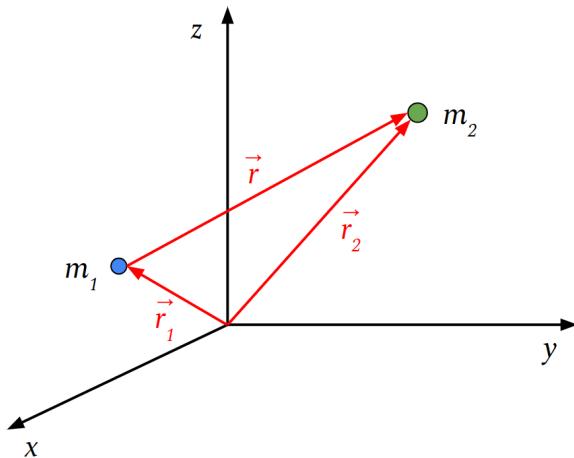


Figura 1 – Posições de duas massas pontuais,  $m_1$  e  $m_2$ .

Fonte: Elaborada pelo autor.

Seja  $\vec{r}$  o vetor posição que liga o corpo 1 ao corpo 2,

$$\vec{r} = \vec{r}_2 - \vec{r}_1, \quad (1)$$

e  $\hat{r}$  o seu versor unitário,

$$\hat{r} = \frac{\vec{r}}{r}, \quad (2)$$

a força  $\vec{F}_{12}$  realizada pelo corpo 2 sobre o corpo 1 e a força  $\vec{F}_{21}$  realizada pelo corpo 1

sobre o corpo 2 são descritas pela lei da gravitação universal,

$$\vec{F} = \frac{Gm_1m_2}{r^2} \hat{r}, \quad (3)$$

com  $\vec{F}_{12} = -\vec{F}_{21}$ .

Pela segunda lei de Newton, escrevemos

$$m_1 \ddot{\vec{r}}_1 = \frac{Gm_1m_2}{r^3} \vec{r} \quad (4)$$

e

$$m_2 \ddot{\vec{r}}_2 = -\frac{Gm_1m_2}{r^3} \vec{r}, \quad (5)$$

que levam às equações de movimento do sistema:

$$\ddot{\vec{r}}_1 = Gm_2 \frac{\vec{r}}{r^3} \quad (6)$$

e

$$\ddot{\vec{r}}_2 = -Gm_1 \frac{\vec{r}}{r^3}. \quad (7)$$

Obtemos duas equações diferenciais ordinárias acopladas e temos como objetivo determinar as posições de cada corpo em função do tempo. Inicialmente, vamos apresentar uma análise da solução analítica dessas equações. Em seguida, apresentamos uma solução numérica de modo a introduzir os métodos computacionais a serem utilizados no decorrer deste trabalho.

### 2.1.1 Solução Analítica

Introduzindo-se o vetor  $\vec{R}$  que representa a posição do centro de massa do sistema,

$$\vec{R} = \frac{m_1 \vec{r}_1 + m_2 \vec{r}_2}{m_1 + m_2}, \quad (8)$$

e utilizando a equação (1), podemos escrever os vetores posição de ambos os corpos como

$$\vec{r}_1 = \vec{R} - \frac{m_2}{m_1 + m_2} \vec{r} \quad (9)$$

e

$$\vec{r}_2 = \vec{R} + \frac{m_1}{m_1 + m_2} \vec{r}. \quad (10)$$

Desse modo, a partir de (9) e (10) em (6) e (7), obtemos as equações

$$\begin{cases} \ddot{\vec{R}} - \frac{m_2}{m_1 + m_2} \ddot{\vec{r}} = Gm_2 \frac{\vec{r}}{r^3} \\ \ddot{\vec{R}} + \frac{m_1}{m_1 + m_2} \ddot{\vec{r}} = Gm_1 \frac{\vec{r}}{r^3} \end{cases} \quad (11)$$

que, quando somadas, nos trazem:

$$(m_1 + m_2) \ddot{\vec{R}} = 0 \quad (12)$$

e

$$\ddot{\vec{r}} = -(m_1 + m_2)G \frac{\vec{r}}{r^3}. \quad (13)$$

Essas equações nos apresentam duas consequências importantes: a primeira é que o centro de massa é um referencial inercial, dada que a derivada segunda de seu vetor posição é zero. A segunda é que o problema de dois corpos se reduziu a uma equação que representa um corpo interagindo gravitacionalmente com o centro de massa do sistema e com sua posição dada pelo vetor  $\vec{r}$ .

Escrevendo o vetor aceleração  $\ddot{\vec{r}}$  em coordenadas polares,

$$\ddot{\vec{r}} = (\ddot{r} - r\dot{\theta}^2)\hat{r} + (2r\dot{\theta} + r\ddot{\theta})\hat{\theta}, \quad (14)$$

e substituindo na equação (13), com  $M = m_1 + m_2$ , obtemos

$$-\frac{GM}{r^2}\hat{r} = (\ddot{r} - r\dot{\theta}^2)\hat{r} + (2r\dot{\theta} + r\ddot{\theta})\hat{\theta}. \quad (15)$$

A componente radial dessa equação nos leva a conservação do módulo do momento angular:

$$2r\dot{\theta} + r\ddot{\theta} = 0 \quad (16)$$

$$\Rightarrow 2r\dot{\theta} + r^2\ddot{\theta} = 0$$

$$\Rightarrow \dot{\theta} \frac{d(r^2)}{dt} + r^2\ddot{\theta} = 0$$

$$\Rightarrow \frac{d(\dot{\theta}r^2)}{dt} = 0.$$

Como  $\dot{\theta}r^2 = \frac{l}{\mu}$ , onde  $\mu$  é a massa reduzida do sistema, temos que  $l$  é constante, consequentemente, o movimento dos corpos com relação ao centro de massa é realizado em um plano.

Além disso, a componente radial da equação (15) nos leva a

$$\ddot{r} - \frac{l^2}{\mu^2 r^3} = -\frac{GM}{r^2}. \quad (17)$$

Realizando-se uma mudança de variável (5), podemos escrever a equação acima como

$$\frac{d^2}{d\theta^2} \left( \frac{1}{r} \right) + \frac{1}{r} = \frac{GM\mu^2}{l^2} \quad (18)$$

cuja solução, em coordenadas polares, será

$$r(\theta) = \frac{\alpha}{1 + \varepsilon \cos \theta}, \quad (19)$$

onde  $\varepsilon$  é a excentricidade e  $\alpha \equiv \frac{l^2}{\mu^2 GM}$  é chamada de semi-*latus rectum* da órbita.

A equação (19) é a equação polar de cônicas e pode representar elipses, hipérboles ou parábolas a depender do valor de  $\varepsilon$ . Para  $\varepsilon = 0$ , temos uma trajetória circular; para  $0 < \varepsilon < 1$  a trajetória será uma elipse com o centro de massa ocupando um dos focos; Para  $\varepsilon = 1$  e  $\varepsilon > 1$  temos trajetórias parabólicas e hiperbólicas, respectivamente. Na Figura 2, temos a representação qualitativa das diferentes seções cônicas obtidas com diferentes valores de  $\varepsilon$ .

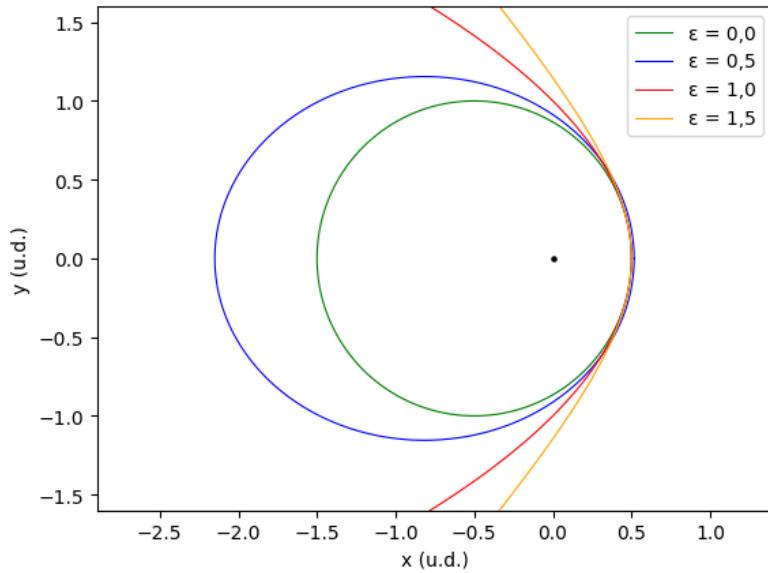


Figura 2 – Diferentes tipos de órbitas para o problema de dois corpos.  $\alpha$  foi selecionado como sendo igual a 1 unidade de distância, mesma unidade dos eixos.

Fonte: Elaborada pelo autor.

### 2.1.2 Solução Numérica

Para a resolução numérica do problema de dois corpos é conveniente resolver as equações (6) e (7) para cada componente do vetor  $\vec{r}_i$  separadamente. Desse modo, escrevemos

$$\ddot{x}_1 = Gm_2 \frac{x_2 - x_1}{r^3}, \quad \ddot{y}_1 = Gm_2 \frac{y_2 - y_1}{r^3}, \quad \ddot{z}_1 = Gm_2 \frac{z_2 - z_1}{r^3} \quad (20)$$

e

$$\ddot{x}_2 = -Gm_1 \frac{x_2 - x_1}{r^3}, \quad \ddot{y}_2 = -Gm_1 \frac{y_2 - y_1}{r^3}, \quad \ddot{z}_2 = -Gm_1 \frac{z_2 - z_1}{r^3}, \quad (21)$$

onde obtemos 6 equações diferenciais ordinárias de segunda ordem acopladas. Para a

resolução, precisaremos de 12 condições iniciais, no qual é mais conveniente utilizar as componentes da posição e velocidade iniciais de cada corpo.

Inicialmente, declaramos um valor para a constante gravitacional  $G$  e definimos as condições iniciais do sistema. Os valores de posição e velocidade dos corpos em cada instante de tempo da simulação serão armazenados em uma estrutura chamada ‘*vetor de estado*’, um *array*  $s$  na forma

$$s = \begin{bmatrix} \vec{r}_1 & \vec{v}_1 \\ \vec{r}_2 & \vec{v}_2 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & z_1 & \dot{x}_1 & \dot{y}_1 & \dot{z}_1 \\ x_2 & y_2 & z_2 & \dot{x}_2 & \dot{y}_2 & \dot{z}_2 \end{bmatrix}$$

a partir do qual utilizaremos para visualizar os dados de cada corpo no decorrer da simulação.

Acompanhado do vetor de estado, outra estrutura importante que utilizaremos será a sua derivada temporal, ou seja, o vetor  $\dot{s}$  dado por

$$\dot{s} = \begin{bmatrix} \dot{\vec{r}}_1 & \dot{\vec{v}}_1 \\ \dot{\vec{r}}_2 & \dot{\vec{v}}_2 \end{bmatrix} = \begin{bmatrix} \dot{x}_1 & \dot{y}_1 & \dot{z}_1 & \ddot{x}_1 & \ddot{y}_1 & \ddot{z}_1 \\ \dot{x}_2 & \dot{y}_2 & \dot{z}_2 & \ddot{x}_2 & \ddot{y}_2 & \ddot{z}_2 \end{bmatrix}$$

que nos auxiliará na resolução numérica de nosso problema.

Após definidas as condições iniciais do sistema, armazenadas em um vetor de estado  $s_0$ , o primeiro passo de nosso algoritmo de resolução é determinar as acelerações de cada corpo, ou seja, as componentes  $\vec{\ddot{v}}_1$  e  $\vec{\ddot{v}}_2$  do vetor  $\dot{s}$  durante um certo incremento de tempo, e isso é feito a partir das equações (6) e (7).

Dadas as velocidades iniciais dos corpos e as acelerações obtidas, podemos construir a estrutura  $s_1$  e integrá-la numericamente com um passo de tempo  $\Delta t$  para obter as próximas posições e velocidades. Realizando-se esse procedimento para um intervalo de tempo desejado, obtemos a evolução temporal das trajetórias de cada corpo. O algoritmo de integração numérica utilizado para isso foi o método de Runge-Kutta de quarta ordem (RK4). A escolha deste método dentre todos os outros abordados no campo do cálculo numérico é a sua simplicidade de implementação, garantindo uma boa precisão e bom custo computacional sem exigir muita complexidade na escrita do código.

Por fim, o último complemento de nosso código para a solução numérica do problema de dois corpos é uma função que computa a energia mecânica total do sistema em cada passo de integração, armazenando esse valor para se ter como estatística ao fim da simulação.

O código foi implementado em *python* e pode ser encontrado na referência (6). Como exemplo de execução, utilizamos as condições iniciais descritas em (7),

$$m_1 = m_2 = 1,0 \times 10^{26} \text{kg}$$

$$s_0 = \begin{bmatrix} 0 & 0 & 0 & 10 & 20 & 40 \\ 3000 & 0 & 0 & 0 & 40 & 0 \end{bmatrix},$$

com unidades de distância em [km], unidades de velocidade em [km/s] e um intervalo de simulação de 0 a 520s. O programa resultou nos gráficos apresentados nas Figuras 3 e 4.

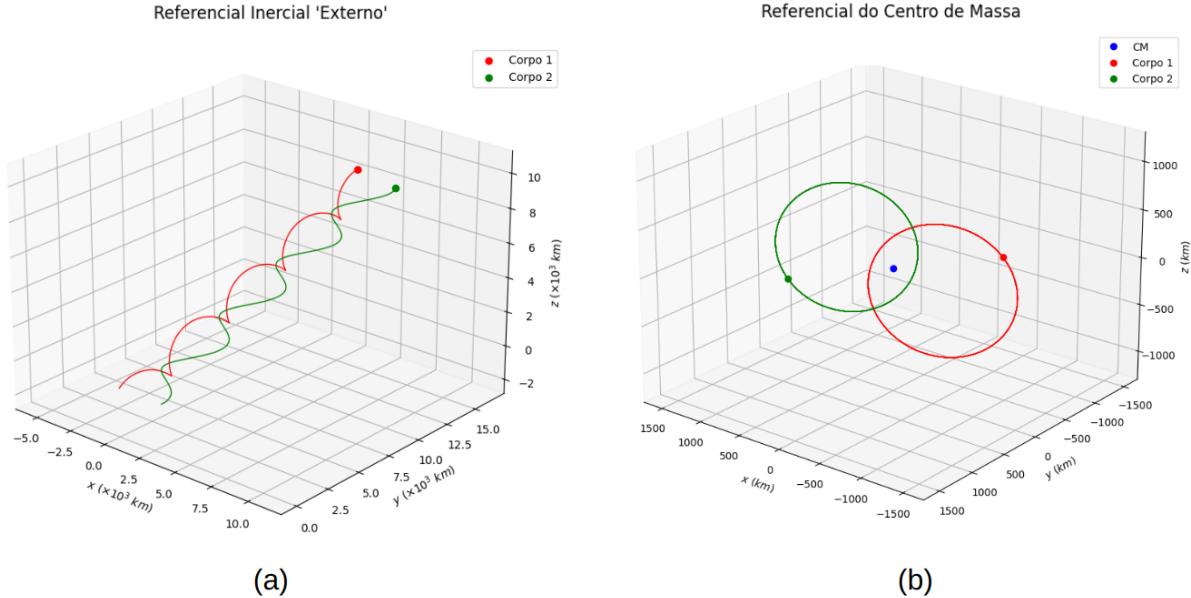


Figura 3 – Resultados da simulação numérica para o problema de dois corpos. (a) Visualização da evolução temporal do sistema em um referencial inercial qualquer. (b) Evolução temporal do sistema no referencial do centro de massa.

Fonte: Elaborada pelo autor.

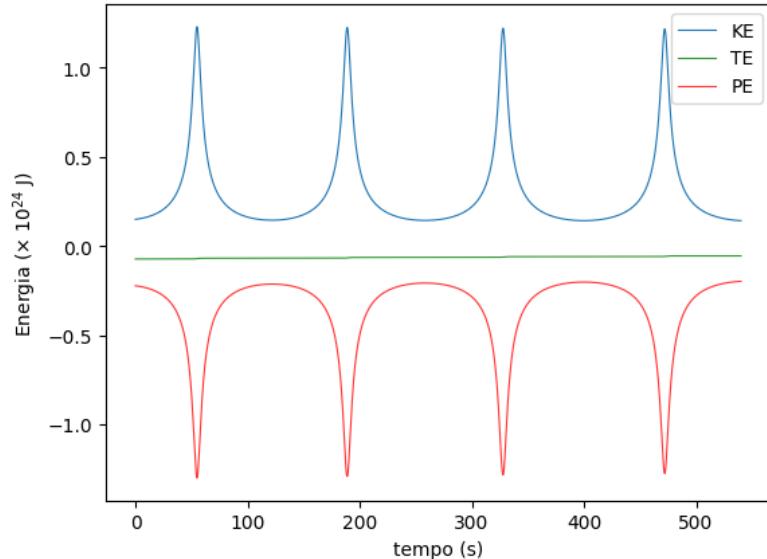


Figura 4 – Análise da conservação da energia mecânica do sistema, onde KE representa a energia cinética, TE, a energia mecânica total e PE, a energia potencial.

Fonte: Elaborada pelo autor.

Na Figura 3a temos a evolução temporal das posições do sistema em um referencial inercial qualquer. Na Figura 3b observamos uma das consequências da equação (19): de que a órbita dos corpos no referencial do centro de massa é realizada em seções de cônicas, neste caso, em elipses. Além disso, na Figura 4 temos a análise da conservação da energia mecânica do sistema.

## 2.2 Problema de Três Corpos

O problema de três corpos surge como uma expansão natural do problema anterior, visto que a maioria dos sistemas físicos interagentes gravitacionalmente, como sistemas planetários, são compostos por um número de corpos maior que dois. Contudo, apesar da presença mais significativa na natureza, tais sistemas já não apresentam solução analítica geral, mesmo para um número de corpos  $n = 3$ , apenas em alguns casos específicos sob suposições especiais a respeito do tipo de movimento e interação (8).

A dificuldade em derivar uma solução geral para este problema está no fato de não existir, até onde sabemos, transformações de coordenadas que simplificam o sistema (4), como o que foi feito para o caso de dois corpos, ao tratarmos o problema no referencial do centro de massa — onde tivemos o trabalho de resolver apenas uma equação diferencial. Em consequência dessa dificuldade, ao longo da história, a implementação de diversos métodos computacionais contribuiu para a resolução do problema e, atualmente, sistemas com  $n > 2$  corpos são analisados com grande precisão numericamente.

A formulação do problema de três corpos parte da lei da gravitação universal de Isaac Newton, apresentada na seção anterior. Consideramos um sistema referencial inercial cartesiano com três corpos de massas  $m_1$ ,  $m_2$  e  $m_3$  com posições dadas pelos vetores  $\vec{r}_1$ ,  $\vec{r}_2$  e  $\vec{r}_3$ , respectivamente, com relação a origem do sistema, como mostrado na Figura 5.

A força resultante que atua em cada corpo de massa  $m_i$  é a soma vetorial das forças que cada corpo de massa  $m_j$  faz sobre ele,

$$\vec{F}_i = \sum_{j \neq i} \vec{F}_{ij}. \quad (22)$$

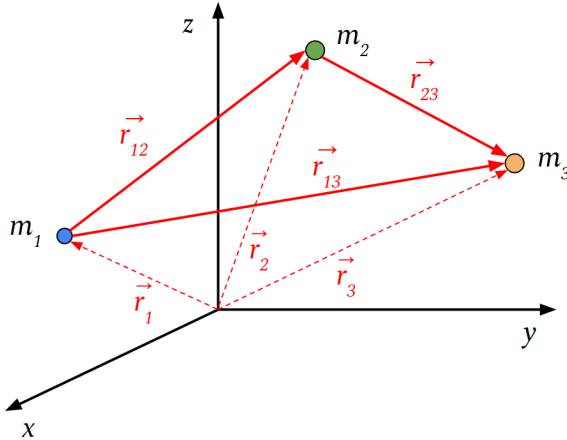
Pela segunda lei de Newton, obtemos a equação de movimento para cada um dos três corpos:

$$m_1 \ddot{\vec{r}}_1 = \frac{Gm_1m_2}{r_{12}^3} \vec{r}_{12} + \frac{Gm_1m_3}{r_{13}^3} \vec{r}_{13}, \quad (23)$$

$$m_2 \ddot{\vec{r}}_2 = -\frac{Gm_2m_1}{r_{12}^3} \vec{r}_{12} + \frac{Gm_2m_3}{r_{23}^3} \vec{r}_{23} \quad (24)$$

e

$$m_3 \ddot{\vec{r}}_3 = -\frac{Gm_3m_1}{r_{13}^3} \vec{r}_{13} - \frac{Gm_3m_2}{r_{23}^3} \vec{r}_{23}. \quad (25)$$

Figura 5 – Posições de três massas pontuais:  $m_1$ ,  $m_2$  e  $m_3$ .

Fonte: Elaborada pelo autor.

Para este caso, vamos abordar apenas a solução numérica. Além disso, vamos simplificar o problema resolvendo-o em um plano, ou seja, apenas as componentes  $x$  e  $y$  de nosso sistema. Desse modo, o vetor de estado será da forma

$$s = \begin{bmatrix} \vec{r}_1 & \vec{v}_1 \\ \vec{r}_2 & \vec{v}_2 \\ \vec{r}_3 & \vec{v}_3 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & \dot{x}_1 & \dot{y}_1 \\ x_2 & y_2 & \dot{x}_2 & \dot{y}_2 \\ x_3 & y_3 & \dot{x}_3 & \dot{y}_3 \end{bmatrix}.$$

Após definidas as condições iniciais do problema, armazenando na estrutura  $s_0$ , semelhante ao abordado na seção anterior, o primeiro passo de nosso *script* é determinar as acelerações de cada corpo a partir das equações (23), (24) e (25). Em seguida, realizamos o procedimento já descrito para determinar as trajetórias dos corpos para cada passo temporal. A implementação do *script* completo pode ser consultada na referência (6).

### 2.2.1 Problema de Três Corpos Restrito

Assim como foi feito na seção anterior, apresentamos um teste de nosso código, em um caso particular do problema de três corpos cuja solução é estável e periódica. As condições iniciais deste exemplo foram retiradas de (9) e o vetor de estado inicial é dado por:

$$s_0 = \begin{bmatrix} -1 & 0 & 0.3471128135672417 & 0.532726851767674 \\ 1 & 0 & 0.3471128135672417 & 0.532726851767674 \\ 0 & 0 & -2 \times 0.3471128135672417 & -2 \times 0.532726851767674 \end{bmatrix}. \quad (26)$$

As trajetórias correspondentes à solução deste problema formam a figura ‘ $\infty$ ’ no plano x-y, como mostrado na Figura 6. Para este sistema, a constante gravitacional foi

definida como  $G = 1$  e foram selecionadas as massas  $m_1$ ,  $m_2$  e  $m_3$  também com valor unitário. Como o valor da constante  $G$  foi alterado, as unidades de massa ( $um$ ), tempo ( $ut$ ) e distância ( $ud$ ) do sistema devem satisfazer

$$\frac{ud^3}{um \cdot ut^2} = \frac{1}{6,6743 \times 10^{-11}} \frac{m^3}{kg \cdot s^2}. \quad (27)$$

O intervalo de simulação utilizado foi de 0 a  $1,5ut$ . Alterando-se este parâmetro, foi observado que os corpos retornam às suas posições iniciais e o movimento começa a se repetir cicличamente após um período  $T = 6,325ut$ , como mostrado na Figura 6b.

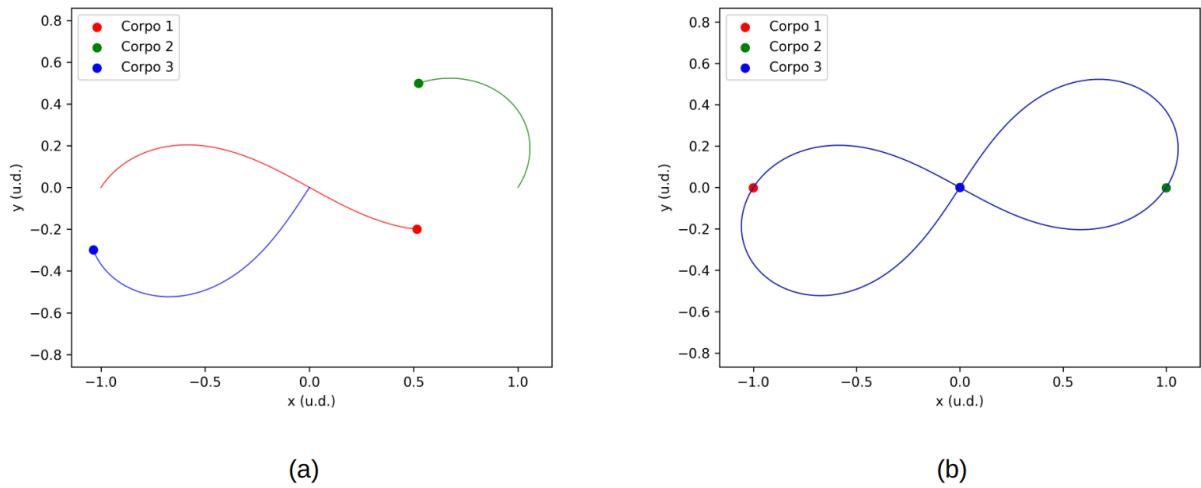


Figura 6 – Resultados da simulação numérica para um problema particular de três corpos.

(a) Visualização das trajetórias do sistema para um intervalo de  $t = 0ut$  à  $t = 1,5ut$ . (b) Visualização das trajetórias do sistema para um período completo,  $t = 0ut$  à  $t = 6,325ut$ .

Fonte: Elaborada pelo autor.

Na Figura 7, temos um gráfico da variação das energias cinética e potencial do sistema durante um período completo, onde podemos observar a conservação da energia mecânica total.

A partir deste exemplo, exploramos uma característica de problemas de  $n$ -corpos com  $n > 2$ : a dependência extremamente sensível à mudanças nas condições iniciais. Para isso, alteramos em 5% o valor da componente  $y$  da velocidade inicial do terceiro corpo. Apenas com essa mudança, as trajetórias passam a apresentar um movimento irregular e sem periodicidade. O gráfico das novas posições em função do tempo é apresentado na Figura 8. Atribuímos esse comportamento ao fato de que qualquer sistema composto por três ou mais corpos interagindo sob a ação de forças não lineares, como a força gravitacional, ao ser submetido a modificações em seu estado em um determinado instante de tempo, passa a apresentar um comportamento caótico (4).

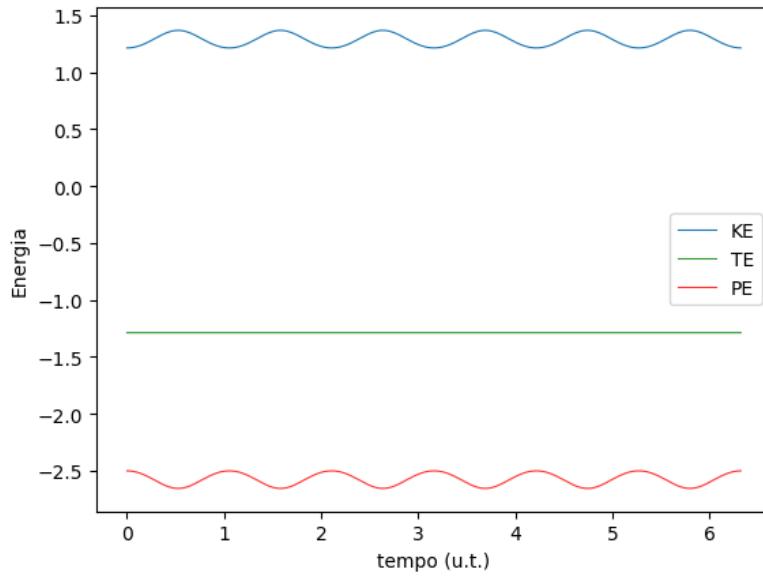


Figura 7 – Análise da conservação da energia mecânica do sistema durante um intervalo de  $t = 0ut$  à  $t = 6,3250ut$ .

Fonte: Elaborada pelo autor.

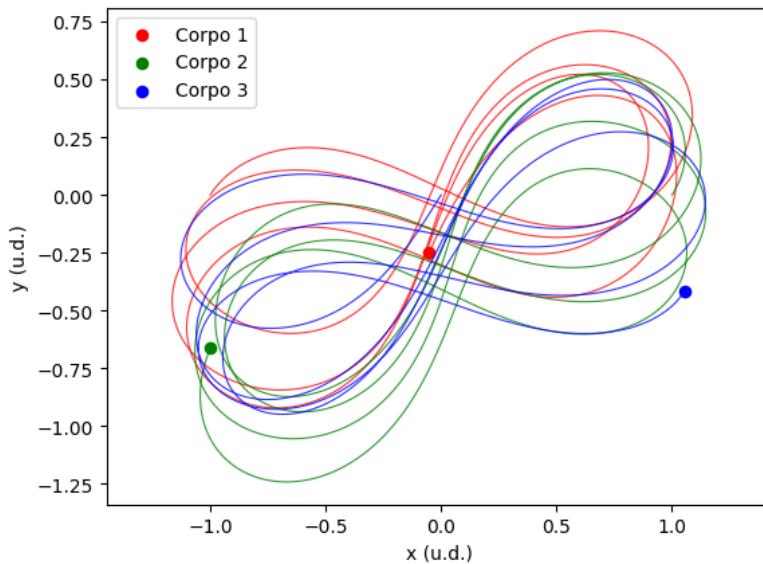


Figura 8 – Trajetórias do sistema composto por três corpos descritos pelas condições iniciais (26) sujeito a um acréscimo de 5% na componente y da velocidade do 3º corpo. O intervalo de simulação utilizado foi de  $t = 0ut$  à  $t = 20ut$ .

Fonte: Elaborada pelo autor.

## 2.3 Simulação de n-corpos

A partir dos códigos anteriores, vemos que algumas estruturas se repetem independente do número de corpos que trabalhamos: como um método para a determinar as forças em cada instante, uma função para a integração numérica das acelerações e velocidades e uma rotina para o cálculo da energia mecânica do sistema. Com base nisso, e

aproveitando-se do paradigma de orientação à objetos da linguagem *python*, foi conveniente armazenar essas estruturas em uma classe, denominada `nBody`, onde a partir dela podemos criar um objeto que recebe como parâmetro de inicialização o número de corpos e possui como métodos internos funções que permitem realizar a simulação. Nesta seção, descrevemos a implementação dessa classe, que generaliza os casos anteriores, realizando uma simulação de n-corpos.

### 2.3.1 Arquitetura da Simulação Implementada

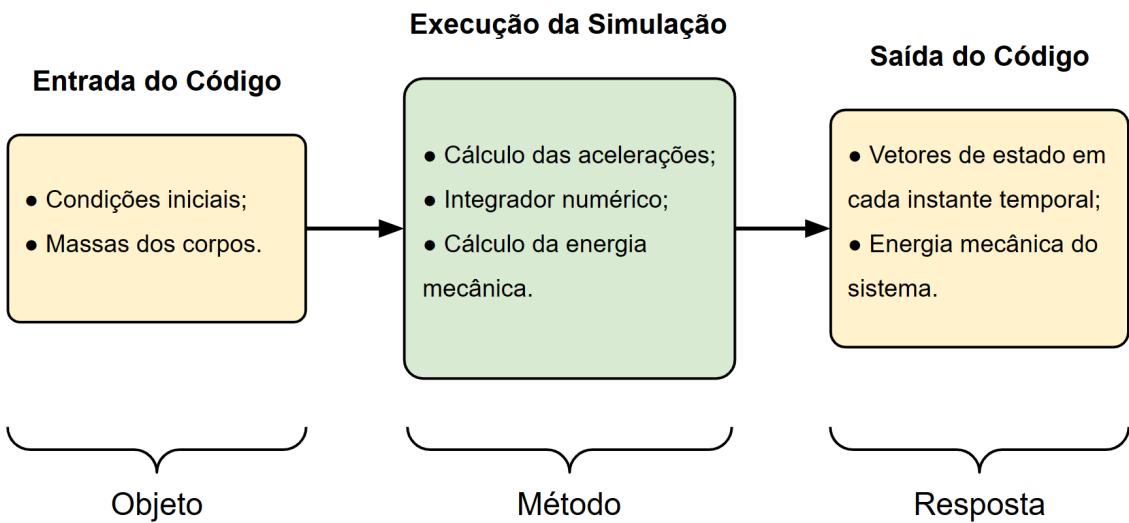


Figura 9 – Esquema de funcionamento do código da simulação de n-corpos.

Fonte: Elaborada pelo autor.

A arquitetura do código desenvolvido é exemplificada no esquema da Figura 9. A primeira parte do algoritmo é a criação do objeto de nossa simulação, que recebe como parâmetro um vetor de estado  $s_0$  com as condições iniciais do sistema, um vetor com os valores de massa de cada corpo e o intervalo de tempo, junto com o passo, utilizado. É nessa etapa que o código identifica o número de corpos que será utilizado na simulação.

O segundo bloco do esquema apresenta a parte mais importante do código: como ele vai executar a simulação em si. Neste trecho, temos uma rotina que possui duas partes principais: um algoritmo que determina as forças e, consequentemente, as acelerações de cada corpo e um integrador numérico que determina as posições e velocidades do sistema em cada instante de tempo. É nessa etapa que os códigos de simulações de n-corpos se diferenciam, pois existem diversos algoritmos para o cálculo das acelerações e diversos integradores numéricos. Neste trabalho, utilizamos o algoritmo de interação “partícula-partícula” (PP) e o integrador descrito anteriormente, RK4.

O algoritmo de interação PP utiliza a equação (22) para determinar a força exercida em cada corpo a partir da soma das forças provocadas por todos os outros corpos do

sistema, sem realizar qualquer tipo de simplificação ou aproximação. É o método mais simples e intuitivo de simulações de n-corpos e retorna com precisão as acelerações de cada partícula. Consequentemente, possui um desempenho computacional que diminui conforme se aumenta o número de corpos, uma vez que para atualizar cada partícula é necessário conhecer as informações de todos as outras, acarretando em um custo  $O(N^2)$ .

No segundo bloco do código, também determinamos as energias cinética, potencial e mecânica total do sistema. Por fim, a última parte do esquema representa a saída do algoritmo. Nessa etapa, o código retorna um *array* tridimensional com as posições e velocidade dos corpos em cada instante de simulação, além de um *array* com as energias do sistema.

### 2.3.2 Suavização Gravitacional

As interações realizadas pela matéria escura em pequenas escalas ainda são um problema em aberto na física. Apesar da existência de diversas hipóteses — como, por exemplo, a aniquilação em raios gama —, ainda não se sabe o comportamento destes corpos conforme se aproximam. Além disso, um ‘problema’ presente nas simulações de n-corpos é que quando a distância entre dois corpos tende a zero, a força gravitacional apresenta uma singularidade, levando a uma divergência na energia cinética do sistema. Tal comportamento não acontece em problemas físicos reais, em razão da atuação de forças que contrapõem a gravitacional para pequenas distâncias.

Com isso, as simulações gravitacionais de n-corpos para o estudo da DM, presentes na literatura da área, utilizam uma estratégia para escapar desses problemas. Introduzindo-se um comprimento de amortecimento (10) na equação geral da força gravitacional entre dois corpos,

$$\vec{F} = \frac{Gm_i m_j}{(r_{ij}^2 + \epsilon^2)^{3/2}} (\vec{r}_i - \vec{r}_j), \quad (28)$$

modificamos o seu valor para pequenas escalas. Para  $\epsilon \ll r_{ij}$  essa equação se aproxima da força gravitacional regular, para  $\epsilon \gg r_{ij}$  a força se torna constante.

Desse modo,  $\epsilon$  limita quão perto as estruturas podem se aproximar de modo a serem tratadas numericamente. A escolha do valor de  $\epsilon$  varia de acordo com a precisão desejada, além dos tipos de interações incluídos na simulação realizada. Assim, introduzi esta modificação na classe `nBody`, generalizando o código para tratar os corpos de maneira acolisional.

## 2.4 O Código GADGET-2

Para testar o desempenho de nosso código e avaliarmos outras métricas relevantes na execução de nossa simulação, decidimos compará-lo com o programa em código aberto

**GADGET-2** (11), comumente utilizado na literatura para simulações cosmológicas de *n*-corpos. Com isso, além de avaliar nosso programa, estudamos um código já otimizado, visto que a otimização de nosso algoritmo não está dentro do escopo deste trabalho, uma vez que ele foi implementado como meio de estudar os conceitos elementares associados às simulações de *n*-corpos.

*GA*laxies with *D*ark *m*atter and *G*as *intEracT* (GADGET) é um código-fonte gratuito utilizado para a realização de simulações cosmológicas e de formação de grandes estruturas a partir de uma abordagem de *n*-corpos. O programa teve sua primeira versão publicada em 2001 (11) e permite também incluir interações de matéria bariônica em suas simulações, a partir do método de hidrodinâmica de partículas suavizadas. A escolha deste código para a utilização no trabalho se deve em razão da sua boa documentação, além de ser *open-source*, tendo sido escrito na linguagem *C* e permitindo modificações de acordo com os interesses de cada simulação.

Ademais, uma grande vantagem de se trabalhar com o **GADGET-2** é o seu desempenho computacional, uma vez que o código é muito bem otimizado por utilizar o recurso de paralelização e um algoritmo em árvore — com custo temporal  $O(N \log N)$  — para executar as simulações.

O algoritmo em árvore é um aprimoramento do algoritmo PP. Nele, as partículas distantes são agrupadas e as contribuições de cada uma para a força total não são calculadas individualmente, mas sim, do grupo como um todo a partir de expansões multipolares em torno do centro de massa do conjunto. Desse modo, o algoritmo divide o espaço em uma árvore onde cada nó corresponde a uma partícula que pode representar um grupo a depender de sua distância. Como a força gravitacional decresce com o quadrado da distância, essas aproximações são fisicamente válidas. O algoritmo controla os erros de precisão através do ângulo de abertura, um parâmetro que diz o quanto pequeno e distante o grupo é para utilizar a aproximação.

#### 2.4.1 Exemplo de Execução: colisão de galáxias espirais

Como exemplo de execução do **GADGET-2**, executamos uma simulação de dinâmica galáctica, cujas condições iniciais descrevem duas galáxias espirais que interagem apenas gravitacionalmente. A distribuição inicial do sistema pode ser encontrada no primeiro quadro da Figura 10 e cada galáxia possui um disco estelar e um halo de matéria escura compostos por 10 mil e 20 mil corpos, respectivamente. As condições iniciais desse sistema foram obtidas juntas com o código e a sua evolução temporal descreve a colisão dessas galáxias.

Executando-se este arquivo de condições iniciais em um intervalo de 0 a  $4 \times 10^9$  anos (um total de 4096 passos de integração), obtivemos as distribuições mostradas na Figura

10. O tempo total de execução foi de apenas 5 minutos e 10 segundo, tendo o código sido executado em computador pessoal com 16GB de memória RAM, um processador 11th Gen Intel Core™ i7-1165G7 @ 2.80GHz × 8 e uma placa de vídeo NVIDIA GeForce GTX 1050. O volume de simulação foi distribuído entre 4 processadores.

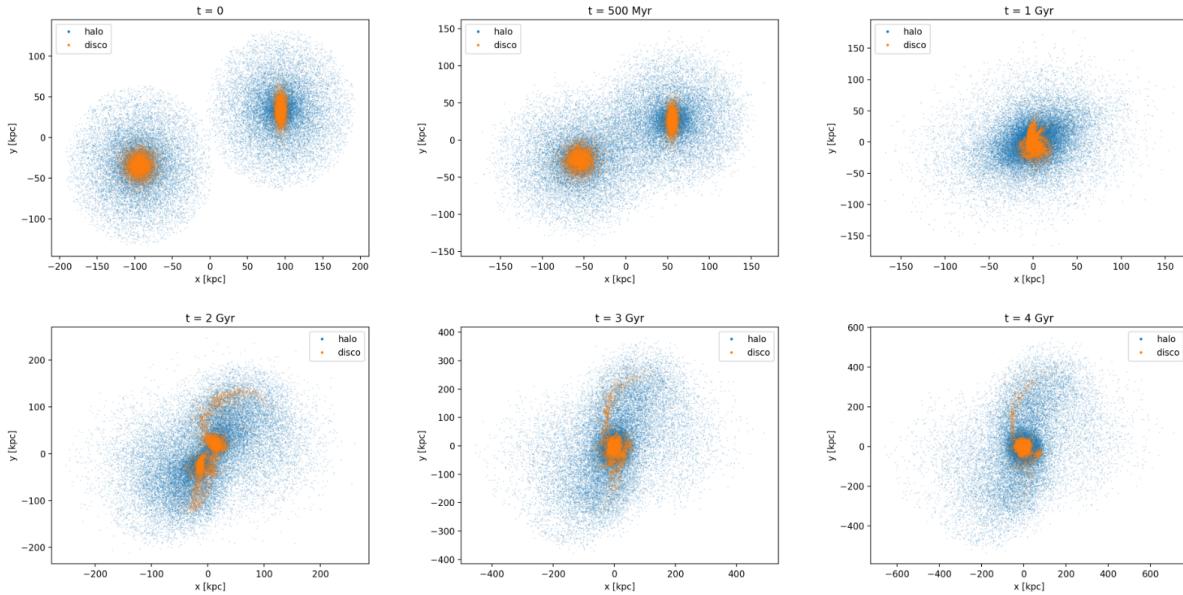


Figura 10 – Evolução da simulação de colisão de duas galáxias espirais interagindo exclusivamente através da gravidade. Na figura, temos um recorte no plano x-y, uma vez que o sistema é tridimensional. Executando-se a simulação por um intervalo de tempo maior, o sistema resultará em uma galáxia elíptica.

Fonte: Elaborada pelo autor.

Após a execução, recriamos a simulação no código de n-corpos escrito em *python* descrito anteriormente. Em virtude da execução deste exemplo ser lenta, e para extrair mais informações, realizamos uma análise do tempo que seria necessário para executar a simulação completa, visando avaliar o desempenho de nosso programa e estimar quais os limites que devem ser atribuídos para as simulações de n-corpos com o algoritmo de interação partícula-partícula. Para isso, executamos a simulação inúmeras vezes variando, em cada uma delas, o número de corpos e registrando os valores de tempo.

Como apresentado por Souza (5), uma métrica que pode ser utilizada para avaliar a precisão de simulações gravitacionais de n-corpos é o erro relativo para a energia mecânica, definido pela equação

$$\delta E = \frac{|E_m(t) - E_m(0)|}{E_m(0)}, \quad (29)$$

onde  $E_m(t)$  e  $E_m(0)$  são os valores da energia mecânica total em um instante  $t$  e no instante inicial para  $t = 0$ . Assim, implementamos essa análise em nosso trabalho.

Além disso, como complemento, avaliamos a dependência do erro relativo em função do número de corpos que utilizamos na simulação. Para isso, semelhante ao descrito

anteriormente, executamos o nosso código variando o número de corpos. Para cada execução, calculamos o erro relativo entre a energia mecânica total nos instantes final e inicial da simulação.

### 3 Resultados

Os resultados da análise do tempo de execução da simulação descrita anteriormente podem ser vistos no gráfico da Figura 11. Através de um ajuste quadrático, uma vez que a evolução temporal segue uma curva  $O(N^2)$ , podemos estimar que o código levaria 7,25 anos para ser executado com todos os 60 mil corpos durante os 4096 passos de simulação.

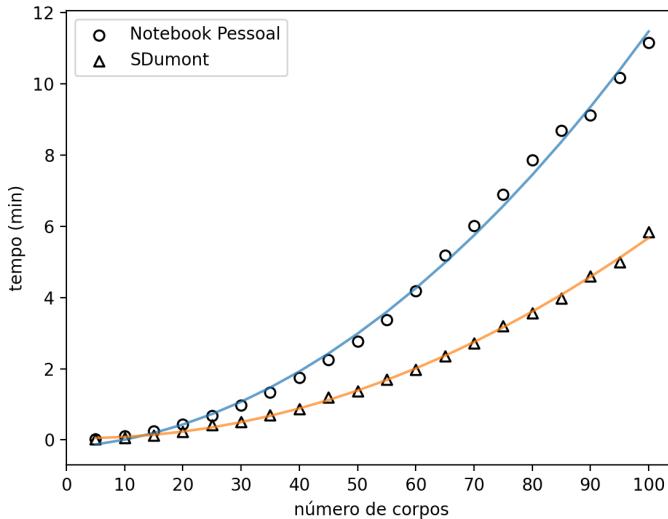


Figura 11 – Tempo de execução em função do número de corpos. No gráfico, temos duas curvas representando a mesma simulação executada em um computador pessoal e no supercluster Santos Dumont. Para a curva azul, o ajuste quadrático foi  $(0, 00106 \pm 0, 00007)x^2 + (0, 011 \pm 0, 008)x - (0, 2 \pm 0, 2)$  e para a curva laranja  $(0, 00059 \pm 0, 00002)x^2 - (0, 003 \pm 0, 002)x + (0, 05 \pm 0, 05)$ .

Fonte: Elaborada pelo autor.

Além do resultado obtido em um computador pessoal, de configurações já descritas acima, a mesma análise foi realizada executando-se o código no supercluster *Santos Dumont* do Laboratório Nacional de Computação Científica (LNCC), distribuindo a simulação entre 4 processadores. Apesar do poder computacional do cluster, a simulação ainda levaria 4,05 anos para ser executada.

Os valores de tempo de execução levados pelo GADGET-2 para executar a mesma simulação não estão descritos no gráfico em razão da abrupta diferença de escala entre os resultados. A partir do gráfico, concluímos que, apesar de mais preciso, o algoritmo PP é muito ineficiente, devendo ser utilizado preferencialmente para o estudo de sistemas com número de corpos inferior a  $10^3$ .

Para a análise da precisão da simulação, na Figura 12a mostramos a evolução temporal do erro relativo para um caso exemplo em que foram utilizados 100 corpos. Vemos que inicialmente o erro relativo cresce até atingir um patamar em que esse valor

estabiliza, apresentando algumas flutuações durante esse período. Tal comportamento se repete nos outros casos em que foram simulados um número diferente de corpos, no qual a partir de um certo ponto o erro relativo se estabiliza em um valor até o final da simulação.

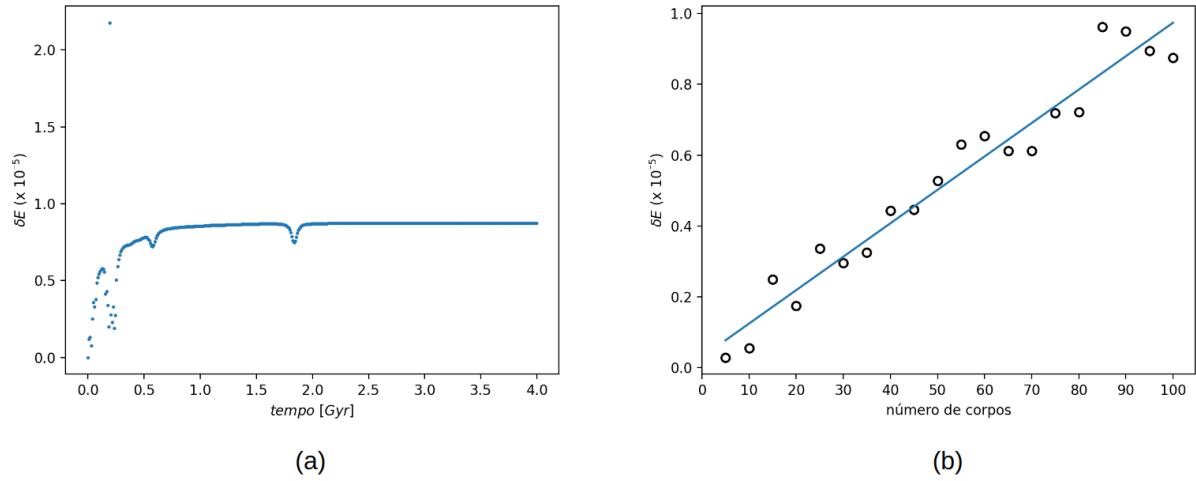


Figura 12 – (a) Erro relativo em função do tempo de execução para 100 corpos. (b) Erro relativo em função do número de corpos simulados. Para a curva mostrada, o ajuste linear foi  $(0,0094 \pm 0,0005)x + (0,03 \pm 0,03)$ , com  $R^2 = 0,95$ .  
Fonte: Elaborada pelo autor.

Além disso, na Figura 12b temos a dependência do erro relativo de acordo com o número de corpos simulados. Vemos no gráfico um comportamento crescente e, se aproximarmos esses pontos para uma reta, obtemos um coeficiente de ajuste  $R^2 = 0,95$ . Utilizando essa curva aproximada, podemos estimar que para os 60 mil corpos utilizados na simulação de colisão de galáxias, apesar da baixa eficiência computacional na execução de nosso programa, o erro relativo ainda seria da ordem de  $10^{-3}$ .

## 4 Conclusão

Neste trabalho, abordamos o paradigma da matéria escura e a sua relação com as simulações de n-corpos. A partir da análise do problema mecânico de dois corpos, apresentamos as soluções analítica e numérica, estendendo, em seguida, o procedimento para sistemas com três corpos. Além disso, descrevemos o desenvolvimento de uma simulação de n-corpos que utiliza como base um algoritmo de interação partícula-partícula para computar a força gravitacional e o método de Runge-Kutta de quarta ordem para a integração numérica. Por fim, comparamos a eficiência de nossa simulação com o código **GADGET-2**, disponível na literatura para simulações cosmológicas.

Como resultado da comparação, inferimos que o algoritmo utilizado para o cálculo da força gravitacional deve ser limitado a sistemas com um número de corpos inferior a  $10^3$ . A partir dessa ordem de grandeza, algoritmos em árvore são recomendados por possuírem uma maior eficiência.

No quesito precisão, avaliamos o erro relativo da energia mecânica total de nosso sistema para um exemplo de condições iniciais que descrevem a colisão de duas galáxias espirais. Variando o número de corpos utilizados na simulação, inferimos que o erro relativo associado a essa simulação, com 60 mil corpos, seria da ordem de  $10^{-3}$ .

Com isso, concluímos a nossa análise sobre as simulações gravitacionais de n-corpos e a sua relação com a matéria escura. Cabe falar que um passo natural que pode ser dado a este trabalho seria o estudo mais aprofundado do código **GADGET-2** e a sua utilização para analisar outros aspectos relacionados a matéria escura: como a formação de grandes estruturas do Universo e a distribuição de matéria escura em galáxias, observando os resultados práticos obtidos através de observações e os obtidos a partir de simulações de n-corpos.

## REFERÊNCIAS

- 1 MARCOMINI, J. A. **Estudo da possibilidade de detecção da matéria escura com telescópios Cherenkov**. 2015. Tese (Doutorado em Ciências) — Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2015.
- 2 OTTO, M. P. **Searching for dark matter in the Galactic Center with the Fermi Gamma-ray Space Telescope**. 2020. Monografia (Graduação) — Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2020.
- 3 NAVARRO, J. F.; FRENK, C. S.; WHITE, S. D. M. The structure of cold dark matter halos. **Astrophysical Journal**, American Astronomical Society, v. 462, p. 563, May 1996. ISSN 1538-4357. DOI: 10.1086/177173.
- 4 SOUZA, J. C. L. **Problema de N Corpos e sua relação com o caos**. 2022. Monografia (Graduação) — Universidade Federal de São Carlos, Sorocaba, 2022.
- 5 SOUZA, A. C. de. **Análise numérica da dinâmica e estabilidade dos problemas de dois e três corpos**. 2001. Dissertação (Mestrado) — Programa de Pós Graduação em Informática, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2001.
- 6 OLEGARIO, M. **Códigos implementados para o problema de n-corpos**. 2023. Disponível em: <https://github.com/marcos-olegario/Undergraduate-Research-Project/tree/main>. Acesso em: 20 maio 2024.
- 7 WEBER, B. **Orbital mechanics and astrodynamics**. 2020. Disponível em: <https://orbital-mechanics.space/intro.html>. Acesso em: 18 março 2023.
- 8 FIGUERÊDO, E.; CASTRO, A. S. d. Um problema de três corpos analiticamente solúvel. **Revista Brasileira de Ensino de Física**, SciELO Brasil, v. 23, p. 289–293, 2001.
- 9 QUARESMA, L. J.; RODRIGUES, M. E. Coreografias no problema de três corpos restrito. **Revista Brasileira de Ensino de Física**, SciELO Brasil, v. 41, 2018.
- 10 AARSETH, S. J.; HOYLE, F. Dynamical evolution of clusters of galaxies, I. **Monthly Notices of the Royal Astronomical Society**, Oxford University Press, v. 126, n. 3, p. 223–255, 1963.
- 11 SPRINGEL, V. The cosmological simulation code gadget-2. **Monthly Notices of the Royal Astronomical Society**, The Royal Astronomical Society, v. 364, n. 4, p. 1105–1134, 2005.