

CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC		
(X) PRÉ-PROJETO	() PROJETO	ANO/SEMESTRE: 2023/2

OTIMIZAÇÃO DE SIMULAÇÕES DE N CORPOS UTILIZANDO INTELIGÊNCIA ARTIFICIAL

Vinícius Manuel Martins

Prof. Aurélio Faustino Hoppe – Orientador

1 INTRODUÇÃO

A simulação computacional de sistemas gravitacionais é um campo de estudo fundamental e antigo na física. Segundo Dehnen e Read (2011), este ramo da pesquisa tem como objetivo modelar o comportamento de partículas em sistemas complexos, como galáxias, estrelas, planetas e outras estruturas cósmicas. Portanto, compreender como a gravidade influencia esses sistemas permite aos cientistas observarem eventos novos, possibilitando a exploração de novas teorias (BAGLA; PADMANABHAN, 2004).

No entanto, a simulação de sistemas com múltiplas partículas é uma tarefa computacionalmente intensiva. Para cada partícula dentro do sistema, é necessário calcular as interações gravitacionais com todas as outras. Conforme destacado por Bagla e Padmanabhan (2004), diversas abordagens matemáticas e interpretações alternativas do problema podem ser empregadas para simplificá-lo. Uma dessas abordagens envolve agrupar as partículas em conjuntos, ou tratar o sistema como um mapa de densidade com vetores de deslocamento (OLIVEIRA *et al.*, 2020).

Segundo Garrison *et al.* (2021), com o avanço da computação, houve um progresso significativo nesse campo, possibilitando a realização de simulações em larga escala. No entanto, muitas dessas melhorias baseiam-se em aproximações dos cálculos, o que pode introduzir imprecisões. Essas imprecisões, de acordo com Gonzalez *et al.* (2020) podem alterar sutilmente as condições iniciais e transformar um sistema que deveria ser determinístico em algo imprevisível a longo prazo, um fenômeno conhecido como caos.

Ainda segundo Garrison *et al.* (2021), a partir deste cenário desafiador e em constante evolução, torna-se necessário a explorar estratégias de otimização para simulações computacionais de sistemas de N partículas, que representam uma classe crucial de sistemas físicos. Tais sistemas são caracterizados pelo estudo do movimento de N partículas que interagem mutuamente por meio da força gravitacional. Além disso, os autores também destacam que a complexidade computacional associada ao cálculo das interações entre todas as partículas em sistemas de N corpos aumenta exponencialmente com o número de partículas, tornando essas simulações demoradas e requerendo bastante recursos computacionais.

Neste contexto, este trabalho tem a seguinte pergunta de pesquisa: Qual é a eficácia das abordagens de otimização, incluindo heurísticas e técnicas baseadas em inteligência artificial, na melhoria da eficiência computacional e da precisão das simulações de sistemas gravitacionais de N partículas, considerando a complexidade intrínseca desses sistemas e a influência do caos em suas dinâmicas?

1.1 OBJETIVOS

Este trabalho tem como objetivo explorar técnicas de otimização para a simulação computacional de sistemas gravitacionais com N partículas, resultando na criação de um artefato computacional.

Os objetivos específicos são:

- a) investigar o uso de heurísticas que possam aprimorar tanto a eficiência quanto a precisão das simulações, permitindo a análise de sistemas cada vez mais complexos e numerosos;
- b) desenvolver uma abordagem baseada em inteligência artificial capaz de prever os movimentos das partículas em simulações de sistemas gravitacionais, mesmo diante da natureza caótica inerente desses sistemas;
- c) disponibilizar visualizações que tornarão as simulações mais acessíveis e informativas, permitindo um entendimento mais profundo e uma exploração mais eficaz dos mistérios gravitacionais do universo;
- d) avaliar a eficiência e a precisão das técnicas de otimização exploradas para simulações de sistemas gravitacionais.

2 TRABALHOS CORRELATOS

Neste capítulo serão apresentados os trabalhos que correlacionam aos objetivos deste trabalho. Para realizar a busca, foi utilizada a plataforma Google Scholar. Os termos de pesquisa utilizados foram: *n-body*

simulation, three body problem, machine learning, optimization e graph network. A busca combinada dessas palavras-chave retornou cerca de 160.00 registros, sendo limitado apenas a artigos criados após 2019. A escolha deles foi feita com base em sua relevância, uma medida do site que leva em conta número de citações, autores e locais de publicação.

Por fim, foram selecionados 4 artigos. A seção 2.1 descreve o uso de redes neurais profundas para o problema de N corpos, explorando seu comportamento em um ambiente caótico (BREEN *et al.*, 2020). Na seção 2.2 são descritos métodos de otimizações e heurísticas utilizados em simuladores (GARRISON *et al.*, 2021). A seção 2.3 aborda o tema por meio de redes neurais orientadas a grafos (GONZALEZ *et al.*, 2020). Por fim, na seção 2.4 será apresentado o desenvolvimento de uma rede neural profunda aplicada a conceitos de mapa de densidade e vetores de deslocamento (OLIVEIRA *et al.*, 2020).

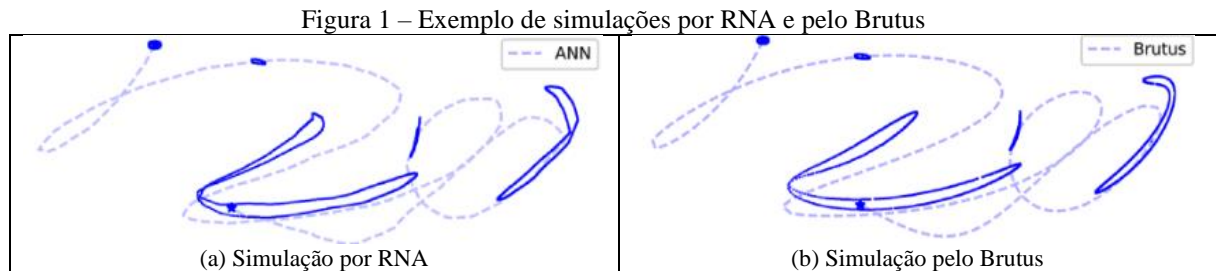
2.1 NEWTON VERSUS THE MACHINE: SOLVING THE CHAOTIC THREE-BODY PROBLEM USING DEEP NEURAL NETWORKS

De acordo com Breen *et al.* (2020), simulações que envolvem três corpos ou mais possuem uma natureza caótica. Dessa forma, pequenas imprecisões nos cálculos podem resultar em resultados completamente distintos. Do ponto de vista computacional, esses problemas apresentam novos desafios, como o uso intensivo de memória e *time-steps* que se aproximam de zero.

Para superar essa questão, Breen *et al.* (2020) propuseram o uso de uma Rede Neural Artificial (RNA) que fosse capaz de prever a trajetória de partículas durante encontros próximos. Eles construíram uma base de dados com 10.000 registros usando Brutus, um integrador numérico para problemas de N corpos. Por uma questão de simplicidade, segundo os autores, essa base de dados foi limitada a problemas envolvendo três corpos com massas iguais e sem velocidades iniciais.

Segundo Breen *et al.* (2020), após treinar a RNA por 1.000 épocas, utilizando um *time-step* de 3.9, obteve-se uma taxa de energia relativa com Mean Absolute Error (MAE) inferior a 0.1%. Apesar disso, em momentos de encontros próximos, onde a energia é altamente sensível à posição, os erros podem chegar a uma média de 10%. Esse problema foi minimizado com a introdução de uma camada de projeção ao modelo. Essa camada, busca preservar a energia do sistema realizando ajustes compensatórios. Com isso, de acordo com Breen *et al.* (2020) conseguiu-se reduzir o MAE para 0.00001%.

A Figura 1 mostra a evolução temporal de uma simulação que não estava incluída nos dados de teste, realizada pela RNA (item a) e pelo Brutus (item b). Os trechos em azul escuro referem-se à trajetória realizada pelos dois métodos em cinco espaços de tempo. Pode-se notar que a RNA conseguiu reproduzir o comportamento caótico desejado por Breen *et al.* (2020).



Fonte: Breen *et al.* (2020).

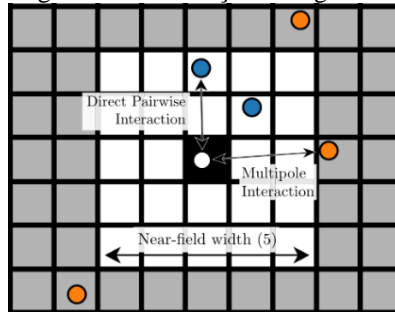
Como resultado, Breen *et al.* (2020) destacam o uso da RNA junto com integradores numéricos. Sugerindo o uso da RNA para realizar as previsões e em seguida usar essa saída como entrada para os integradores tradicionais. Isso permite, segundo os autores, eliminar gargalos ao mesmo tempo que mantém um alto nível de precisão. Além disso, Breen *et al.* (2020) ressaltam que o estudo explorar a resolução do problema eliminando algumas limitações, como a igualdade de massas entre os corpos, resultando em uma solução mais generalista para problemas de três corpos.

2.2 THE ABACUS COSMOLOGICAL N-BODY CODE

Garrison *et al.* (2021) desenvolveram o Abacus, um código de alta precisão e performance para a simulação cosmológica de N corpos. Segundo os autores, utilizando diferentes métodos matemáticos, em uma arquitetura orientada a eventos otimizada para o uso de unidades de processamento gráficas (GPUs), é possível realizar o processamento de milhões de partículas com uma perda na ordem de 1.6×10^{-4} . Além disso, Garrison *et al.* (2021) também implementaram estratégias de gerenciamento de memória, como o agrupamento das partículas em *clusters* e o uso otimizado do disco. Segundo os autores, essas melhorias foram projetadas para que pudessem ser executadas em *hardware* mais acessível.

A Figura 2 demonstra o cenário de otimização processado pelo algoritmo. Nele, a força resultante em uma partícula é determinada por duas componentes, interações de campo próximo e campo distante. Segundo Garrison *et al.* (2021), interações de campo próximo são mais custosas, mas valores mais precisos são cruciais para o resultado da simulação. Em contrapartida, as interações de campo distante são calculadas por meio de aproximações, visto que imprecisões nesse cenário não representam muita perda na solução final.

Figura 2 – Otimização no algoritmo



Fonte: Garrison *et al.* (2021).

Como resultado, Garrison *et al.* (2021) demonstraram o desempenho do Abacus em uma simulação que utilizou 44 núcleos de unidade central de processamento (CPU) e 6 GPUs NVIDIA V100, alcançando o processamento de 60 trilhões de partículas, em 97 diferentes condições iniciais, a uma taxa de processamento superior a 70 milhões de partículas por segundo.

Por fim, Garrison *et al.* (2021) destacam que as evoluções no *hardware* de computadores são feitas buscando resolver problemas em diversos níveis (aprendizado de máquina, renderizações gráficas e análises de dados). Além disso, segundo os autores, o Abacus fez isso utilizando mudanças na arquitetura do código em alto nível, combinada como otimizações matemáticas, o uso personalizado da memória possibilitando uma *pipeline* orientada a eventos, entre outras coisas. Para Garrison *et al.* (2021), esse tipo de abordagem, torna a solução encontrada escalável para computadores do futuro, dado a sua característica altamente paralelística.

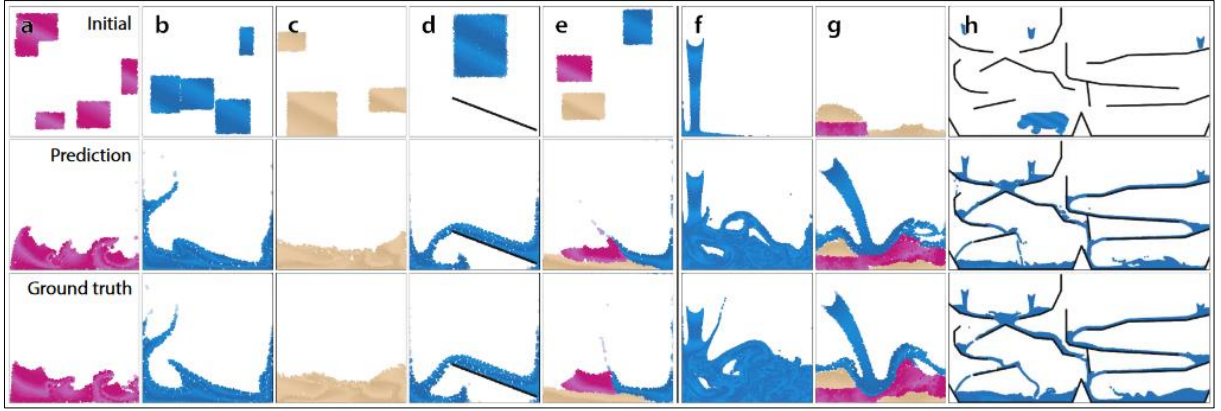
2.3 LEARNING TO SIMULATE COMPLEX PHYSICS WITH GRAPH NETWORKS

Gonzalez *et al.* (2020), desenvolveram um *framework* de machine learning orientado a grafos que denominaram “Simuladores orientados a redes de grafos”. Segundo os autores, ele é composto por três partes principais: (i) o *encoder*, (ii) o *processor* e (iii) o *decoder*. O *encoder*, cria um grafo onde cada nó é uma partícula do sistema. As conexões do nó são criadas entre partículas que estejam dentro de um “raio de conectividade” e a cada *time-step*, essas ligações são recalculadas utilizando o algoritmo do vizinho mais próximo. O peso de cada aresta é a distância relativa entre os dois nós. Já o *processor*, realiza o processamento das interações entre os nós. Ele é composto por M camadas, onde em cada uma delas ocorre a passagem de mensagens entre os nós. Além disso, são utilizadas conexões residuais, permitindo que a saída de uma camada possa ser a entrada de outra, pulando algumas camadas intermediárias. Por fim, o *decoder*, é uma *perceptron* multicamadas que extrai a aceleração das partículas a partir do grafo resultante. Com a aceleração, é possível realizar a integração de Euler para extrair a velocidade e posição das partículas e enfim atualizar o sistema.

Para o treinamento, Gonzalez *et al.* (2020) utilizaram partículas com diferentes propriedades, como água, areia e gosma, e exploraram as interações entre elas. O *dataset* foi gerado utilizando ferramentas e simuladores já existentes, como o BOXBATH, SPlisHSPlasH e o Taichi-MPM. Com isso, Gonzalez *et al.* (2020) notaram que alguns hiperparâmetros impactavam diretamente no resultado, tais como: (i) o número de trocas de mensagens, (ii) o número de conexões (arestas do grafo) e (iii) o ruído inicial. Esse último, os autores destacam que foi uma estratégia adotada na etapa de treinamento para mitigar a acumulação de erros nos cálculos no momento da validação.

Na Figura 3 apresenta os resultados do estudo de forma visual. Cada coluna representa um cenário de validação (não incluídos na fase de treinamento), sendo que na primeira linha estão as condições iniciais, na segunda estão as previsões da Graph Networks Simulate (GNS) e na última a base de verdade. As partículas rosas são gosma, as azuis são água e as beges são areia. Os obstáculos em preto se referem a matéria sólidos estáticos. Pode-se notar, que o modelo conseguiu entender bem a dinâmica entre os diversos materiais, conseguindo realizar previsões plausíveis a longo prazo.

Figura 3 – Exemplo de previsão do GNS



Fonte: Gonzalez *et al.* (2020).

Como resultado, Gonzalez *et al.* (2020) obtiveram simulações com o Mean Squared Error (MSE) na ordem de 10^{-3} , variando de acordo com a escolha dos materiais, sua interação, etc. Com isso, os autores constataram que o modelo conseguiu generalizar bem o problema e entender a natureza dos materiais, além dos comportamentos físicos envolvidos. Em comparação com outros modelos, o GNS se mostrou mais simples e exato, com espaço para otimizações com o uso de computação paralela.

2.4 FAST AND ACCURATE NON-LINEAR PREDICTIONS OF UNIVERSES WITH DEEP LEARNING

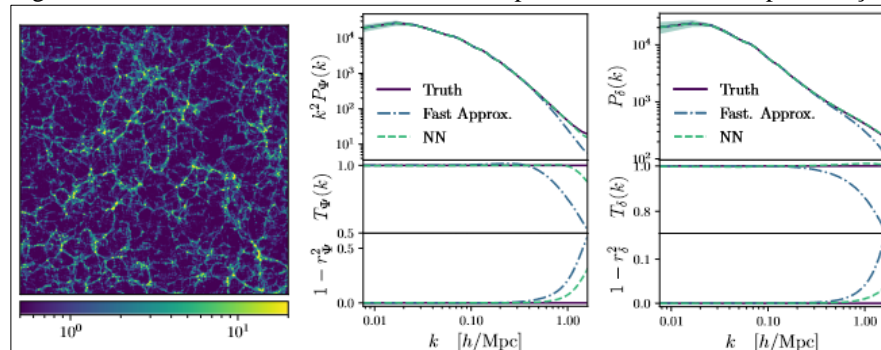
Oliveira *et al.* (2020) abordaram o problema das aproximações em simulações de N corpos em escalas menores. Nestes cenários, segundo os autores, a flutuação da densidade das partículas geralmente se torna significativa, onde pequenas imprecisões podem invalidar as previsões.

Em busca de alternativas para enfrentar esse desafio, Oliveira *et al.* (2020) desenvolveram uma *Convolutional Neural Network* (CNN) com a arquitetura V-NET que recebe como entrada os vetores de deslocamento das partículas e retorna o vetor de deslocamento resultante. Segundo os autores, a arquitetura trabalha com três resoluções diferentes do problema, com duas camadas de *downsampling* e duas de *upsampling*. Além disso, são utilizadas conexões residuais em cada uma das camadas de convolução.

De acordo com Oliveira *et al.* (2020), para o treinamento do modelo, foram utilizadas 210 simulações de N corpos do projeto Quijote. Dessa amostragem, 180 foram utilizados para treino, 20 para validação e 10 para teste. Cada simulação possui cerca de 512^3 partículas dispostas dentro de um cubo com volume de 1 (Gpc/h)^3 . Cada um dos universos simulados emprega 4 conjuntos distintos de parâmetros cosmológicos, sendo que um deles se assemelha muito às condições do nosso próprio universo.

Oliveira *et al.* (2020) utilizaram a função de perda MSE, ao qual foi aplicada tanto ao vetor resultante quanto à quantidade de partículas em cada *voxel*, que representa uma célula do *grid* no domínio da simulação. Segundo os autores, o modelo obteve uma acurácia superior aos métodos de aproximação tradicionais, com mais de 99% de acurácia e um tempo de treinamento duas vezes mais rápido. A Figura 4 apresenta os resultados obtidos. O painel à esquerda se refere ao campo de densidade esperado, conforme simulado pelo projeto Quijote. Os gráficos do meio e da direita, se referem aos resultados da rede neural considerando respectivamente, o vetor resultante e a densidade de partículas nos *voxels*. A partir da Figura 4, pode-se observar que conforme k (flutuações na densidade) aumenta, os métodos de aproximação tradicionais falham em prever os valores, em contraste com o modelo proposto por Oliveira *et al.* (2020).

Figura 4 – Gráficos dos resultados obtidos comparando a métodos de aproximação



Fonte: Oliveira *et al.* (2020).

Como conclusão, Oliveira *et al.* (2020) ressaltam que o estudo abre caminho para um maior aproveitamento científico de futuras missões astronômicas. Além disso, afirmam que tais melhorias podem possibilitar futuros cientistas realizarem simulações complexas do universo, com alta precisão a partir do seu computador pessoal.

3 PROPOSTA DO ARTEFATO COMPUTACIONAL

Neste capítulo será descrita a justificativa do estudo, definindo os requisitos funcionais e não funcionais e as metodologias abordadas.

3.1 JUSTIFICATIVA

O Quadro 1 apresenta uma comparação dos quatro trabalhos correlatos selecionados, destacando suas principais características. A disposição do quadro é a seguinte: cada linha representa uma característica distinta e cada coluna um trabalho correlato.

Quadro 1 – Comparativo entre os trabalhos correlatos

Trabalhos Correlatos Características	Breen <i>et al.</i> (2020)	Garrison <i>et al.</i> (2021)	Gonzalez <i>et al.</i> (2020)	Oliveira <i>et al.</i> (2020)
Natureza do problema	Simulações de três corpos com natureza caótica	Simulação cosmológica de N corpos	Simulações de interação de partículas	Simulações de N corpos em escalas gravitacionais menores
Estratégia de otimização	Uso de uma RNA para prever as trajetórias	Otimizações no algoritmo, estratégias de paralelismo e métodos matemáticos	Rede neural orientada a grafos	Uso de uma CNN com arquitetura V-Net
Base utilizada	10.000 registros gerados pelo Brutus	Não se aplica	Dados gerados com diferentes ferramentas: BOXBATH, SPLisHSPlasH e Taichi-MPM	Dados obtidos a partir das simulações do projeto Quijote
Métricas de desempenho	MAE inferior a 0.1%	Perca na ordem de 1.6×10^{-4}	MSE na ordem de 10^{-3}	Mais de 99% de acurácia
Resultados	RNA reproduziu natureza caótica de forma precisa	Processamento de 70 milhões de partículas por segundo	Previsões precisas e uma solução mais simples de se trabalhar	Modelo mais preciso e rápido que métodos tradicionais

Fonte: elaborado pelo autor.

A partir do Quadro 1, pode-se notar que todos os trabalhos enfrentam desafios computacionais significativos relacionados à simulação de sistemas de N partículas. Breen *et al.* (2020), Garrison *et al.* (2021) e Oliveira *et al.* (2020) focam em sistemas de N corpos gravitacionais, abordando o problema em diferentes escalas. Por outro lado, Gonzalez *et al.* (2020) exploram a simulação de partículas interagindo, considerando suas diferentes propriedades. Todos os problemas possuem complexidades intrínsecas e buscam explorar características únicas do tema.

Quanto às estratégias de otimização, os trabalhos divergem em suas abordagens. Breen *et al.* (2020) propõem o uso de uma RNA para prever trajetórias em simulações de três corpos. Garrison *et al.* (2021) desenvolvem o código Abacus, buscando uma maior otimização para GPUs, visando simulações cosmológicas em grande escala. Gonzalez *et al.* (2020) introduzem o GNS, para prever interações de partículas. Oliveira *et al.* (2020) utilizam uma CNN com arquitetura V-NET em escalas menores.

Em relação aos resultados, todos os trabalhos alcançaram melhorias significativas. Breen *et al.* (2020) conseguiram simular a natureza caótica do sistema com sua RNA, enquanto Garrison *et al.* (2021) conseguiram alto desempenho com o Abacus em simulações cosmológicas em larga escala. Gonzalez *et al.* (2020) demonstraram previsões precisas das interações de partículas com o GNS. Oliveira *et al.* (2020) destacaram a precisão e velocidade superior de sua CNN em comparação com métodos tradicionais.

Com isso, pode-se observar que os quatro trabalhos abordam desafios de simulações de N partículas de maneiras distintas, utilizando estratégias de otimização variadas e obtendo melhorias consideráveis em eficiência e precisão. Cada abordagem contribui para o avanço da pesquisa nesse campo, destacando a versatilidade de técnicas computacionais e de aprendizado de máquina para lidar com problemas complexos de física e cosmologia.

Neste trabalho serão exploradas diferentes técnicas e abordagens para melhorar a eficiência das simulações de sistemas de N corpos. Isso pode envolver o uso de algoritmos de aproximação que reduzem o número de interações a serem calculadas, o desenvolvimento de técnicas de paralelização para acelerar o processamento em sistemas de alto desempenho, ou até mesmo a aplicação de inteligência artificial para prever o comportamento das partículas de forma mais eficiente. Além disso, será avaliada a precisão das simulações após a otimização, uma vez que é importante garantir que as simplificações ou aproximações não comprometam a qualidade dos resultados. O trabalho também se concentrará em criar visualizações e ferramentas que tornem as simulações mais acessíveis e informativas, facilitando a interpretação dos resultados e a tomada de decisões em contextos científicos e de engenharia. Por fim, este trabalho busca contribuir para o avanço no campo das simulações de sistemas de N corpos, explorando estratégias de otimização que permitam simulações mais rápidas e eficientes, mantendo a precisão dos resultados. Isso tem implicações significativas em diversas áreas do conhecimento e pode abrir portas para uma compreensão mais profunda de sistemas complexos regidos pela gravidade.

3.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O artefato computacional a ser desenvolvido deverá:

- a) processar uma simulação utilizando forças gravitacionais dado as configurações iniciais para as partículas, incluindo posições, massas, velocidades iniciais, *time-step* e outras propriedades relevantes (RF – Requisito Funcional);
- b) explorar o uso de redes neurais e outras heurísticas de otimizações (RF);
- c) incluir estratégias de otimização para reduzir o tempo de processamento e minimizar a demanda de recursos computacionais, especialmente em simulações em larga escala (RF);
- d) disponibilizar métricas de execução após a execução de cada teste (como tempo de execução e consumo de memória) (RF);
- e) gerar visualizações gráficas das simulações, facilitando a compreensão dos resultados e permitindo a análise visual (RF);
- f) ser implementada utilizando as linguagens Go e Python (RNF – Requisitos Não Funcional);
- g) deve ser capaz de lidar com simulações de sistemas de N partículas com eficiência, proporcionando tempos de execução razoáveis mesmo para simulações complexas (RNF).

3.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- a) revisão bibliográfica: busca de conceitos gerais de simulação computacional, aprendizado de máquina, heurísticas de otimização e trabalhos correlatos;
- b) levantamento dos requisitos: baseando-se nas informações da etapa anterior, reavaliar os requisitos propostos para a aplicação;
- c) definição das ferramentas para modelagem e armazenamento das simulações: pesquisar e escolher as ferramentas mais apropriadas para a modelagem e armazenamento das simulações;
- d) definição das técnicas de otimização: pesquisar e (re)definir as técnicas/algoritmos que serão utilizadas na otimização da simulação de N partículas;
- e) definição do cenário de simulação: a partir do item (d) estabelecer o cenário de simulação, a escala do sistema (N) e os níveis de precisão;
- f) construção da base de dados: utilizar integradores numéricos para gerar uma base de dados e utilizar bases de dados já existentes, como a do projeto Quijote;
- g) implementação do simulador: desenvolvimento de uma simulação com a solução mais simples (menos ótima) utilizando Go, exportando os dados simulados para um arquivo;
- h) implementação da visualização: desenvolvimento de uma visualização dessas simulações utilizando Python e a biblioteca Pygame;
- i) otimizações no simulador: explorar estratégias que possam aumentar a capacidade de processamento do simulador, como aproximações e heurísticas;
- j) implementação do modelo de inteligência artificial: desenvolver um modelo em Python capaz de prever as forças envolvidas em uma simulação de N corpos, levando em consideração seu comportamento caótico;
- k) validação dos resultados: realizar uma comparação dos resultados obtidos a partir da solução mais simples, com otimizações e a prevista pelo modelo de inteligência artificial. Além disso, apontar a eficiência, precisão, pontos positivos e negativos das soluções pesquisadas.

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 – Cronograma de atividades a serem realizadas

etapas / quinzenas	2024									
	fev.		mar.		abr.		maio		jun.	
	1	2	1	2	1	2	1	2	1	2
revisão bibliográfica										
levantamento dos requisitos										
definição das ferramentas para modelagem e armazenamento das simulações										
definição das técnicas de otimização										
definição do cenário de simulação										
construção da base de dados										
implementação do simulador										
implementação da visualização										
otimizações no simulador										
implementação do modelo de inteligência artificial										
validação dos resultados										

Fonte: elaborado pelo autor.

4 REVISÃO BIBLIOGRÁFICA

Este capítulo descreve brevemente sobre os assuntos que fundamentarão o estudo a ser realizado: conceitos de simulação de N corpos, heurísticas de otimização e aprendizado de máquina.

Segundo Bagla (2005), as simulações de N corpos são uma abordagem fundamental na modelagem computacional de sistemas gravitacionais complexos, onde N representa o número de partículas envolvidas. Este conceito implica no cálculo das trajetórias e energia de cada partícula no sistema ao longo do tempo. No entanto, existem desafios significativos, como a escalabilidade computacional e a precisão numérica, especialmente quando se lida com muitas partículas (SHARP, 2006). Assim como, a complexidade algorítmica aumenta exponencialmente com o número de corpos, demandando estratégias eficientes para otimização e aceleração computacional (SPURZEM, 2001).

Segundo Kofler *et al.* (2014), heurísticas de otimização desempenham um papel crucial na melhoria do desempenho das simulações de N corpos, proporcionando métodos eficazes para lidar com a complexidade computacional inerente. Essas técnicas são essenciais para otimizar a performance temporal e espacial das simulações, permitindo uma rápida convergência para soluções aceitáveis (WINKER; GILLY, 2004). No entanto, os desafios incluem a seleção apropriada de heurísticas para diferentes cenários, assim como uma boa parametrização que permita maior desempenho em ambientes dinâmicos e altamente interativos (KARIMI-MAMAGHAN *et al.*, 2022).

Para Wu e Meng (2020), o aprendizado de máquina é uma forma eficiente de realizar uma análise estatística em cima de um conjunto de dados, permitindo a identificação de padrões. A incorporação dessas técnicas representa uma extensão promissora para aprimorar as simulações de N corpos (CARLEO, 2019). Os desafios envolvidos incluem a necessidade de conjuntos de dados representativos, a correta interpretação dos modelos gerados e a adaptação contínua a cenários complexos (L'HEUREUX *et al.*, 2017).

REFERÊNCIAS

- BAGLA, Jasjeet Singh; PADMANABHAN, Thanu. Cosmological N-body simulations. **Springer Link**, Berlim, v. 49, n. 2, p. 161–192, 26 nov. 2004.
- BAGLA, Jasjeet Singh. Cosmological N-Body simulation: Techniques, Scope and Status. **Current Science Association**, Bangalore, v. 88, n. 7, p. 1088–100, 10 abr. 2005.
- BREEN, Philip G. *et al.* Newton versus the machine: solving the chaotic three-body problem using deep neural networks. **Monthly Notices of the Royal Astronomical Society**, Oxford, v. 494, n. 2, p. 2465–2470, 22 abr. 2020.
- CARLEO, G. *et al.* Machine learning and the physical sciences. **Reviews of Modern Physics**, Riverdale, v. 91, n. 4, 6 dez. 2019.
- DEHNEN, Walter; READ, Justin. N-body simulations of gravitational dynamics. **The European Physical Journal Plus**, v. 126, n. 5, maio 2011.
- GARRISON, Lehman H. *et al.* The abacus cosmological N-body code. **Monthly Notices of the Royal Astronomical Society**, Oxford, v. 508, n. 1, p. 575–596, 7 set. 2021.
- GONZALEZ-SANCHEZ, Alvaro *et al.* **Learning to Simulate Complex Physics with Graph Networks**. Ithaca, [2020]. Disponível em: <https://arxiv.org/abs/2002.09405>. Acesso em: 16 set. 2023.

- KARIMI-MAMAGHAN, Maryam *et al.* Machine Learning at the service of Meta-heuristics for solving Combinatorial Optimization Problems: A state-of-the-art. **European Journal of Operational Research**, Leeds, v. 296, n. 2, p. 393-422, 2022.
- KOFLER, Klaus *et al.* Kd-Tree Based N-Body Simulations with Volume-Mass Heuristic on the GPU. **IEEE International Parallel & Distributed Processing Symposium Workshops**, Phoenix, p. 1256-1265, 1 mai. 2014.
- L'HEUREUX, Alexandra *et al.* Machine Learning With Big Data: Challenges and Approaches. **IEEE Access**, Piscataway, v. 5, p. 7776-7797, 2017.
- OLIVEIRA, Renan Alves de *et al.* Fast and Accurate Non-Linear Predictions of Universes with Deep Learning. **arXiv (Cornell University)**, 2 dez. 2020.
- SHARP, Philip W. N-body simulations: The performance of some integrators. **ACM Transactions on Mathematical Software**, New York, v. 32, n. 3, p. 375-395, set. 2006.
- SPURZEM, Rainer. Astrophysical N-Body Simulations: Algorithms and Challenges. **Astrophysics and Space Science Library**, Dordrecht, v. 263, p. 49-58, 1 jan. 2001.
- WINKER, Peter; GILLI, Manfred. Applications of optimization heuristics to estimation and modelling problems. **Springer**, Dordrecht, v. 47, n. 2, p. 211-223, set. 2004.
- WU, Hao; MENG, Fan Ju. Review on Evaluation Criteria of Machine Learning Based on Big Data. **Journal of Physics: Conference Series**, Bristol, v. 1486, n. 5, p. 052026, 1 abr. 2020.