UNIVERSIDADE FEDERAL DO RIO GRANDE - FURG
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação de Mestrado

LipidEase: Ferramenta para Automatização de Simulações de Bicamadas Lipídicas no GROMACS

Yago Mendes Paes

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande - FURG, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientador: Prof. Dr. Karina dos Santos Machado
Co-orientador: Prof. Dr. Adriano Velasque Werhli

Rio Grande, 2026

86

P126l Paes, Yago Mendes

LipidEase: ferramenta para automatização de simulações de bicamadas lipídicas no GROMACS / Yago Mendes Paes – 2025.

116 f.

Dissertação (Mestrado) – Universidade Federal do Rio Grande – Programa de Pós-Graduação em Computação, 2025.

Orientadora: Dra Karina dos Santos Machado
Co-orientador: Dr. Adriano Velasque Werhli

1. Computação. 2. Dinâmica molecular . 3. Bicamada lipídica 4. Simulação computacional. 5. GROMACS . I. Machado, Karina dos Santos. II. Werhli, Adriano Velasque. III. Título.

CDU 004

Catalogação na fonte: Bibliotecária Brenda Sequeira CRB10/1979

ATA DE SESSÃO DE DEFESA DE DISSERTAÇÃO DE MESTRADO

Ata nº 12/2025

Aos trinta e um dias do mês de outubro de dois mil e vinte e cinco, às quatorze horas, ocorreu a sessão de Defesa de Dissertação de Mestrado de **Yago Mendes Paes**, que apresentou a dissertação intitulada **"LipidEase: Ferramenta para Automatização de Simulações de Bicamadas Lipídicas no GROMACS"**, realizada sob orientação da Profa. Dra. Karina dos Santos Machado e coorientação do Prof. Dr. Adriano Velasque Werhli. A banca examinadora foi constituída pelos Profa. Dra. Hélida Salles Santos (FURG) e Prof. Dr. Leonardo Henrique França de Lima (UFSJ), sob a presidência da orientadora. Após a apresentação do trabalho, a banca arguiu o candidato e, a seguir, deliberou pela

(X) aprovação da Dissertação
( ) aprovação da Dissertação, sugerindo modificações no texto
( ) reprovação da Dissertação

Rio Grande, 31 de outubro de 2025.

gov.br Documento assinado digitalmente
HELIDA SALLES SANTOS
Data: 18/11/2025 09:23:39-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. Hélida Salles Santos

gov.br Documento assinado digitalmente
LEONARDO HENRIQUE FRANCA DE LIMA
Data: 10/11/2025 13:44:24-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Leonardo Henrique França de Lima

gov.br Documento assinado digitalmente
KARINA DOS SANTOS MACHADO
Data: 05/11/2025 21:49:19-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. Karina dos Santos Machado
Orientadora

AGRADECIMENTOS

Agradeço à Universidade Federal do Rio Grande (FURG), em especial ao Programa de Pós-Graduação em Computação, por me proporcionar um ambiente de excelência e por todas as oportunidades que solidificaram minha formação como mestre. Tenho imenso orgulho de ter trilhado mais esta etapa da minha jornada acadêmica nesta instituição.

À minha orientadora, Professora Dra. Karina dos Santos Machado, expresso minha mais profunda gratidão. Por ter, mais uma vez, me acolhido e orientado, dando continuidade à jornada que iniciamos na graduação. Sua paciência — aparentemente inesgotável — e sabedoria foram a base para a construção deste trabalho.

Agradeço a todos os integrantes do Combi-Lab, minha comunidade científica e fonte constante de inspiração. A caminhada tornou-se mais leve e estimulante ao lado de pessoas tão inteligentes e dedicadas (às vezes). As discussões e trocas de experiências foram essenciais para o meu aprendizado em diversos aspectos da vida.

À minha família, meu pilar, por todo o incentivo e apoio incondicional. Vocês sempre garantiram que eu tivesse as melhores condições para me dedicar aos estudos e à pesquisa. Sem o carinho, a compreensão e até mesmo os puxões de orelha, eu não teria chegado até aqui. Obrigado por me tornarem alguém melhor.

Aos meus amigos — Rato, Monge, João Pedro, Letícia, Ana, Amanda, Cecília, Laura e Marina —, que me conhecem como as palmas das próprias mãos e, talvez por isso, foram meu alicerce inabalável durante toda esta jornada. Sou profundamente grato por todo o apoio, por cada conversa e ombro amigo que me ofereceram. Em cada momento de dúvida, cansaço ou surto, vocês foram minha rede de segurança e meu lembrete de quem eu sou para além desta dissertação. Mais do que incríveis, vocês foram (e são) essenciais. Cada um, com seu jeito particular e insubstituível, manteve-me são e tornou essa caminhada muito mais leve. Vocês são o meu maior privilégio.

A todos que, direta ou indiretamente, fizeram parte desta etapa da minha vida, o meu muito obrigado.

RESUMO

PAES, Yago Mendes. **LipidEase: Ferramenta para Automatização de Simulações de Bicamadas Lipídicas no GROMACS**. 2026. 118 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

A dinâmica molecular (DM) é uma técnica computacional essencial para investigar sistemas biológicos, como membranas celulares, cujas bicamadas lipídicas desempenham papéis cruciais. Contudo, a configuração e execução de simulações de DM de bicamadas lipídicas com o GROMACS são processos trabalhosos, propensos a erros e que demandam conhecimento técnico considerável, impactando a reprodutibilidade e a eficiência da pesquisa. Este trabalho apresenta a ferramenta computacional LipidEase, desenvolvida em Python, que automatiza o fluxo completo de preparação e execução de simulações de DM de bicamadas lipídicas no GROMACS. O LipidEase tem uma interface gráfica (GUI) intuitiva para entrada de parâmetros e orquestra automaticamente a construção do sistema — incluindo a definição da caixa de simulação, a formação de mono e bicamadas, a solvatação e a adição de íons —, além da execução sequencial das etapas de simulação (minimização de energia, equilíbrios NVT e NPT com restrições posicionais e fase de produção), atualizando dinamicamente os arquivos de topologia. Entre suas funcionalidades-chave, destacam-se a geração automática de arquivos de proveniência detalhados, alinhados aos princípios FAIR para rastreabilidade, e a criação de pacotes de simulação autocontidos, prontos para execução em ambientes de computação de alto desempenho (HPC). A funcionalidade da ferramenta foi validada por meio da geração e simulação bem-sucedida de nove sistemas distintos de bicamadas lipídicas, incluindo fosfolipídios padrão (como DPPC e POPC) e sistemas contendo o lipídio PIM_2, tanto puro quanto em complexo com fármacos. A estabilidade estrutural dos sistemas foi confirmada por análises quantitativas (RMSD, área por lipídio) e inspeção visual das trajetórias. Espera-se que o LipidEase contribua significativamente para a simplificação, padronização e acessibilidade de simulações de bicamadas lipídicas, reduzindo o tempo de preparação, minimizando erros e aumentando a reprodutibilidade desses estudos computacionais.

Palavras-chave: Dinâmica molecular, Bicamada lipídica, Simulação Computacional, GROMACS.

ABSTRACT

PAES, Yago Mendes. **LipidEase: Tool for the Automation of Lipid Bilayer Simulations in GROMACS**. 2026. 118 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

Molecular Dynamics (MD) is an essential computational technique for investigating biological systems such as cell membranes, whose lipid bilayers play crucial roles. However, configuring and executing MD simulations of lipid bilayers using GROMACS is a laborious process, prone to errors, and requires considerable technical expertise, impacting both reproducibility and research efficiency. This work presents LipidEase, a computational tool developed in Python that automates the entire workflow of preparing and executing MD simulations of lipid bilayers in GROMACS. LipidEase features an intuitive graphical user interface (GUI) for parameter input and automatically orchestrates system construction — including simulation box definition, mono- and bilayer formation, solvation, and ion addition — as well as the sequential execution of simulation stages (energy minimization, NVT and NPT equilibration with positional restraints, and production), dynamically updating topology files. Key functionalities include the automatic generation of detailed provenance files, aligned with the FAIR principles for data traceability, and the creation of self-contained simulation packages ready for execution in High-Performance Computing (HPC) environments. The functionality of the tool was validated through the successful generation and simulation of nine distinct lipid bilayer systems, including standard phospholipids (such as DPPC and POPC) and systems containing the lipid PIM_2, both pure and complexed with drugs. Structural stability was confirmed through quantitative analyses (RMSD, area per lipid) and visual inspection of simulation trajectories. LipidEase is expected to significantly contribute to the simplification, standardization, and accessibility of lipid bilayer simulations, reducing setup time, minimizing errors, and enhancing the reproducibility of these computational studies.

Keywords: Molecular dynamics; Lipid bilayer; Computational simulation; GROMACS..

LISTA DE FIGURAS

1 Exemplo da composição de uma membrana biológica.Fonte: Elaborado pelo autor (2025) 22
2 Estruturas de bicamada formadas por moléculas anfipáticas. (A) Bicamada lipídica. (B) Vesícula lipossomal. Fonte: Elaborado pelo autor (2025) 23
(a) 23
(b) 23
3 Desenho esquemático da execução de uma dinâmica molecular, ilustrando o ciclo de cálculo de forças e atualização das posições atômicas. Fonte: Elaborado pelo autor (2025) 25
4 Escala de tempo aproximada de modelagem molecular nos diferentes tamanhos dos sitemas: Quântico, *all-atom*, *coarse-grained* e mesoescala. Fonte: Elaborado pelo autor (2025) 28
5 Diagrama de componentes da arquitetura do LipidEase, detalhando os principais módulos e suas interações no fluxo de trabalho da ferramenta. Fonte: Elaborado pelo autor (2025) 53
6 Fluxograma computacional da ferramenta LipidEase. As cores representam as fases principais: (1) Inicialização e Preparação (amarelo), (2) Construção do Sistema (azul/vermelho) e (3) Execução da Dinâmica (roxo/verde). Fonte: Elaborado pelo autor (2025) 57
7 Interface gráfica desenvolvida em *Python* com *Tkinter* para configuração dos parâmetros de simulação (Aba 1 - Seleção de parâmetros da simulação). 66
8 Interface gráfica desenvolvida em *Python* com *Tkinter* (Aba 2 - Solvatação e Neutralização do sistema). 68
9 Interface gráfica desenvolvida em *Python* com *Tkinter* (Aba 3 - Execução dos passos de dinâmica molecular, geração de *script* para *cluster*). 70
10 Primeira etapa executada pela ferramenta: Definição dos vetores da caixa de simulação para um lipídio. 72
11 Criação da monocamada lipídica com 64 lipídios, replicando o primeiro lipídio nos eixos X e Y, na quantidade definida pelo usuário. . . 72
12 Concatenação dos arquivos de monocamadas para formação da bicamada lipídica, com espaçamento de 0,1 nm entre elas no eixo Z. Contém diferença de representação visual para enfatizar as diferentes estruturas. 73

13 Bicamada com 128 moléculas de lipídios PIM_2 construída. 74
14 Caixa de água superior na espessura definida por usuário (eixo Z), do tamanho da bicamada nos eixos X e Y através da obtenção de coordenadas máximas e mínimas do sistema. A caixa de simulação está deslocada abaixo, em azul, para demonstrar a posição original da caixa de água, antes do translado. 75
15 Sistema de bicamada lipídica construído e minimizado, com solvente e íons presentes. Os íons foram aumentados durante a visualização para confirmar sua presença no sistema. 76
16 Finalização da primeira etapa de equilíbrio: NVT. Nota-se que os lipídios possuem restrição de movimento nos átomos de fósforo, mantendo a posição central deles no eixo Z durante essa etapa. 77
17 Finalização da segunda etapa de equilíbrio: NPT. Nota-se que os lipídios possuem restrição de movimento nos átomos de fósforo, mantendo a posição central deles no eixo Z durante essa etapa. 78
18 Estrutura final após a execução da dinâmica molecular, sem restrição de movimentos no sistema, permitindo a livre movimentação de todas as moléculas. 78
19 Evolução do sistema PIM_2 puro: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 79
20 Gráficos de análise para o sistema PIM_2 puro. 80
21 Evolução do sistema PIM_2 com Benzohidroxamato: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 81
22 Gráficos de análise para o sistema PIM_2 com Benzohidroxamato. . . 82
23 Evolução do sistema PIM_2 com Nitrobenzohidroxamato: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 83
24 Gráficos de análise para o sistema PIM_2 com Nitrobenzohidroxamato. 84
25 Evolução do sistema DPPC: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 86
26 Gráficos de análise para o sistema DPPC. 87
27 Evolução do sistema DLPE: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 88
28 Gráficos de análise para o sistema DLPE. 89
29 Evolução do sistema DMPC: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 90
30 Gráficos de análise para o sistema DMPC. 91
31 Evolução do sistema DOPC: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 92
32 Gráficos de análise para o sistema DOPC. 93
33 Evolução do sistema POPC: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 94
34 Gráficos de análise para o sistema POPC. 95
35 Evolução do sistema POPE: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM). 96
36 Gráficos de análise para o sistema POPE. 97

37 Arquivo `lipid_provenance.json` e parâmetros salvos. 116
38 Arquivo `lipid_provenance.json` e parâmetros salvos. 117

39 Arquivo `lipid_provenance.json` e parâmetros salvos. 117
40 Arquivo `lipid_provenance.json` e parâmetros salvos. 117
41 Arquivo `lipid_provenance.json` e parâmetros salvos. 117

LISTA DE TABELAS

1	Comparação entre os principais programas de dinâmica molecular. . .	32
2	Comparação entre os principais programas de dinâmica molecular. . .	32
3	Comparação entre os programas e a ferramenta desenvolvida - Parte 1	44
4	Comparação entre os programas e a ferramenta desenvolvida - Parte 2	45
5	Principais formatos de arquivo utilizados no GROMACS e na ferramenta LipidEase.	51
6	Requisitos Funcionais da ferramenta LipidEase.	52
7	Requisitos Não Funcionais da ferramenta LipidEase.	53
8	Comparação entre o LipidEase e o CHARMM-GUI para a construção de bicamadas lipídicas.	64
9	Parâmetros de simulação para os seis sistemas de lipídios comuns utilizados na validação da ferramenta LipidEase.	71
10	Resultados da análise de área por lipídio (A_{pl}) para cada sistema. . . .	99

LISTA DE ABREVIATURAS E SIGLAS

AA	*All-Atom*
ADMIRAL	Automated Dynamics, Molecular Interaction, and Reporting Analysis
AMBER	Assisted Model Building with Energy Refinement
Apl	Área por lipídio (*Area per lipid*)
BH	Benzohidroxamato
CG	*Coarse-Grained*
CGenFF	*CHARMM General Force Field*
CHARMM	*Chemistry at HARvard Macromolecular Mechanics*
CLI	*Command Line Interface*
CPU	*Central Processing Unit*
DFT	*Density Functional Theory*
DM	Dinâmica Molecular
DMPC	Dimiristoilfosfatidilcolina
DMQ	Dinâmica Molecular Quântica
DNA	Ácido Desoxirribonucleico
DOPC	Dioleoilfosfatidilcolina
DLPE	Dilaurilfosfatidiletanolamina
DPPC	Dipalmitoilfosfatidilcolina
FAIR	*Findable, Accessible, Interoperable, Reusable*
FEP	*Free Energy Perturbation*
GPU	*Graphics Processing Unit*
GROMACS	*GROningen MAchine for Chemical Simulations*
GUI	*Graphical User Interface*
HPC	*High-Performance Computing*
IA	Inteligência Artificial
JSON	*JavaScript Object Notation*

LAMMPS	*Large-scale Atomic/Molecular Massively Parallel Simulator*
LLM	*Large Language Model*
MD	*Molecular Dynamics*
MDCrow	*Molecular Dynamics Crow*
MDP	Arquivo de Parâmetros de Simulação no GROMACS
MM-PBSA	*Molecular Mechanics Poisson-Boltzmann Surface Area*
NAMD	*Nanoscale Molecular Dynamics*
NBH	Nitrobenzohidroxamato
NPT	*Ensemble Isotérmico-Isobárico*
NVT	*Ensemble Canônico*
PIM_2	Fosfatidil-mio-inositol dimanosídeo
POPC	Palmitoil-oleoil-fosfatidilcolina
POPE	Palmitoil-oleoil-fosfatidiletanolamina
RF	Requisito Funcional
RNA	Ácido Ribonucleico
RNF	Requisito Não Funcional
RMSD	*Root Mean Square Deviation*
RMSF	*Root Mean Square Fluctuation*
VMD	*Visual Molecular Dynamics*

SUMÁRIO

1 Introdução **16**
1.1 Objetivos . . . 18
1.1.1 Objetivo geral . . . 18
1.1.2 Objetivos específicos . . . 18
1.2 Motivação . . . 19
1.3 Organização . . . 20

2 Referencial teórico **21**
2.1 Membranas Biológicas . . . 21
2.1.1 Bicamadas Lipídicas . . . 22
2.1.2 Membranas Biológicas e Fármacos . . . 23
2.2 Bioinformática . . . 24
2.3 Dinâmica Molecular . . . 24
2.3.1 Interações moleculares e equações fundamentais em Dinâmica Molecular 26
2.3.2 Condições periódicas de contorno (Periodic Boundary Conditions – PBC) 27
2.3.3 Níveis de Simulação por Dinâmica Molecular . . . 27
2.3.4 Softwares para Dinâmica Molecular . . . 29
2.3.5 Comparação entre os pacotes de simulação molecular . . . 31
2.3.6 Dinâmica Molecular e Busca de Fármacos . . . 32
2.3.7 DM de Lipídios - *GROMACS* . . . 34
2.4 Automação de Processos Científicos . . . 36
2.4.1 Benefícios da Automação na Bioinformática e Dinâmica Molecular . . . 37
2.5 Reprodutibilidade Científica . . . 37

3 Trabalhos Relacionados **39**
3.1 MDCrow . . . 39
3.2 ADMIRAL . . . 40
3.3 GMX-QK . . . 41
3.4 Packmol . . . 41
3.5 CHAPERONg . . . 42
3.6 EasyAmber . . . 43
3.7 SAMSON Connect e *GROMACS* Wizard . . . 43
3.8 Comparação entre as Ferramentas Apresentadas . . . 44
3.8.1 Comparação entre o LipidEase e o CHARMM-GUI . . . 46

4 Materiais e Métodos **48**
4.1 Linguagem de programação *Python* . . . 48
4.2 Formatos de Arquivos . . . 49
4.2.1 Arquivos de Estrutura Atômica (.pdb e .gro) . . . 49
4.2.2 Arquivos de Topologia (.top e .itp) . . . 49
4.2.3 Arquivos de Parâmetros de Simulação (.mdp) . . . 49
4.2.4 Arquivos de Campo de Força (.ff) . . . 50
4.2.5 Arquivos de Entrada e Saída do Pré-processador (.tpr) . . . 50
4.2.6 Arquivos de Energia e Trajetória (.edr, .trr, .xtc) . . . 50
4.2.7 Arquivos de Restrição de Posição . . . 50
4.2.8 Arquivos de Proveniência e Relatórios (.json e .log) . . . 51
4.2.9 Arquivos de Compactação e Backup . . . 51
4.2.10 Resumo dos Formatos Utilizados . . . 51
4.3 Levantamento de requisitos . . . 52
4.4 Arquitetura da Ferramenta . . . 53
4.5 Processo de desenvolvimento . . . 54
4.6 LipidEase: Funcionalidades da ferramenta . . . 55
4.6.1 Interface e Pré-processamento de Dados de Entrada . . . 56
4.6.2 Geração da Bicamada Lipídica . . . 59
4.6.3 Execução dos Passos da Dinâmica Molecular . . . 60
4.6.4 Justificativa dos tempos de equilíbrio . . . 60
4.6.5 Saída e Pós-Processamento . . . 61
4.6.6 Reprodutibilidade e rastreabilidade . . . 61
4.6.7 Funcionalidades, vantagens e limitações . . . 62

5 Resultados **63**
5.1 CHARMM-GUI x LipidEase . . . 63
5.2 Interface Gráfica . . . 64
5.2.1 Aba 1: Definição de Parâmetros (*Parameter Definition)* . . . 64
5.2.2 Aba 2: Montagem do Sistema (*System Assembly)* . . . 67
5.2.3 Aba 3: Dinâmica Molecular (*Molecular Dynamics*) . . . 67
5.3 Validação da Ferramenta com Múltiplos Sistemas Lipídicos . . . 71
5.4 Estudo de caso 1: Simulações com lipídio PIM_2 . . . 71
5.4.1 Caixa de simulação estrutura lipídica . . . 71
5.4.2 Criando monocamadas . . . 72
5.4.3 Concatenação das monocamadas . . . 73
5.4.4 Sistema de bicamada pronto . . . 73
5.4.5 Solvatação . . . 74
5.4.6 Protocolo de equilíbrio e simulação . . . 75
5.4.7 PIM_2 . . . 79
5.4.8 PIM_2 com Benzohidroxamato . . . 81
5.4.9 PIM_2 com Nitrobenzohidroxamato . . . 83
5.5 Estudos de caso 2: Simulações com seis lipídios comuns . . . 85
5.5.1 Propriedades estruturais de bicamadas lipídicas . . . 85
5.5.2 DPPC . . . 85
5.5.3 DLPE . . . 88
5.5.4 DMPC . . . 90
5.5.5 DOPC . . . 92

5.5.6 POPC 94
5.5.7 POPE 96

6 Conclusão **100**

Referências **103**
6.1 Anexo 1 — Pseudocódigos do LipidEase 112
6.1.1 Algoritmo 1 – Execução Principal 112
6.1.2 Algoritmo 2 – Estrutura da Classe App 112
6.1.3 Algoritmo 3 – Salvamento de Parâmetros 113
6.1.4 Algoritmo 4 – Montagem e Solvatação 113
6.1.5 Algoritmo 5 – Neutralização com Íons 113
6.1.6 Algoritmo 6 – Geração dos Arquivos .mdp 113
6.1.7 Algoritmo 7 – Execução das Etapas de Dinâmica 114
6.1.8 Algoritmo 8 – Cálculo das Dimensões do Lipídio 114
6.1.9 Algoritmo 9 – Distribuição de Lipídios 114
6.1.10 Algoritmo 10 – Combinação das Monocamadas 114
6.1.11 Algoritmo 11 – Sistema Solvatado 114
6.1.12 Algoritmo 12 – Atualização da Topologia 115
6.1.13 Algoritmo 13 – Registro de Proveniência 115

7 Anexo B **116**
7.1 Arquivo de proveniência 116

1 INTRODUÇÃO

A descoberta e o desenvolvimento de fármacos são processos complexos que demandam compreensão detalhada das interações moleculares em sistemas biológicos [78, 97]. Dentre esses sistemas, as membranas celulares desempenham papel central, atuando como barreiras seletivas e mediadoras de interações entre compostos bioativos e o ambiente intracelular [77, 92]. Estruturalmente, as membranas são formadas por bicamadas lipídicas, arranjos dinâmicos de moléculas anfipáticas, como fosfolipídios, que organizam-se espontaneamente em ambientes aquáticos [58]. Essas estruturas não apenas definem a arquitetura celular, mas também influenciam diretamente a farmacocinética de substâncias terapêuticas, desde a difusão até a ligação com alvos proteicos [71].

A dinâmica molecular surge como uma técnica computacional essencial para estudar tais sistemas em escala atômica [79, 99]. A dinâmica molecular (DM) é uma técnica computacional que utiliza princípios da mecânica clássica para prever o movimento de átomos e moléculas ao longo do tempo. Por meio de equações baseadas nas leis de Newton e campos de força que descrevem as interações entre as partículas, é possível simular o comportamento de sistemas biológicos em escala atômica, observando fenômenos como flexibilidade estrutural, interações entre fármacos e membranas e estabilidade de sistemas lipídicos [28]. Programas como GROMACS [5], AMBER [19] e NAMD [73] são amplamente utilizados para essas simulações, mas exigem configuração manual extensiva, além de conhecimento específico de áreas como física, química e biologia para parametrização, especialmente para sistemas complexos como bicamadas lipídicas. A preparação de arquivos de entrada (estruturas moleculares, topologias, parâmetros de força) e a execução sequencial de etapas (minimização de energia, equilíbrio termodinâmico e simulação de produção) são propensas a erros humanos e consomem tempo valioso, limitando a escalabilidade e reprodutibilidade de estudos [8].

Nesse contexto, a automação de workflows em DM torna-se essencial. Ferramentas como ADMIRAL [10] e GMX-QK [84] demonstram avanços na simplificação de simulações, principalmente focadas em proteínas e complexos proteína-ligante, mas lacunas persistem para sistemas lipídicos. A construção de bicamadas no GROMACS, por exemplo, envolve comandos manuais para etapas críticas que demandam conheci-

mento técnico da ferramenta para execução do procedimento. Além disso, a incompatibilidade entre formatos de arquivos de diferentes softwares restringe a interoperabilidade e a análise pós-simulação [4].

Apesar dos avanços recentes, a ausência de fluxos automatizados voltados especificamente para bicamadas lipídicas ainda representa um desafio significativo. A necessidade de realizar etapas manualmente implica em barreiras técnicas e riscos operacionais que podem comprometer a qualidade e a reprodutibilidade das simulações.

As simulações de DM envolvem várias etapas, começando pela preparação do sistema, seguida da minimização de energia, que serve para eliminar contatos estéricos indesejados. A seguir, o sistema passa pela fase de equilíbrio sob diferentes condições termodinâmicas controladas (*ensembles* estatísticos), como o NVT (mantendo constantes Número de partículas, Volume e Temperatura) e o NPT (mantendo constantes Número de partículas, Pressão e Temperatura). Essa etapa visa garantir que o sistema esteja termodinamicamente estável antes de iniciar a fase principal da simulação (etapa de produção), onde os dados de interesse são coletados [32, 38]. Ao longo desse processo, é preciso definir vários parâmetros, como o campo de força, as condições de periodicidade (limites da caixa de simulação), os algoritmos que controlam temperatura e pressão, e o tempo de simulação. Consequentemente, o processo gera um volume significativo de dados, como trajetórias atômicas e energias. Por conta da complexidade e da sensibilidade desses resultados — que podem variar conforme os parâmetros ou as versões dos programas usados —, é essencial o registro de todas as informações sobre a simulação. Esse detalhamento, conhecido como manutenção da proveniência dos dados [35], é fundamental para garantir a reprodutibilidade, um princípio básico da ciência. Adicionalmente, contribui para a Ciência Aberta (*Open Science*), alinhando-se aos princípios FAIR: que os dados sejam Encontráveis, Acessíveis, Interoperáveis e Reutilizáveis [100].

Diante dos desafios expostos, que englobam desde a complexidade técnica até as barreiras para a reprodutibilidade científica [8, 42], esta dissertação apresenta o desenvolvimento e a validação da ferramenta LipidEase. Essa ferramenta visa automatizar a orquestração das etapas de preparação e execução de simulações de dinâmica molecular de bicamadas lipídicas com o software GROMACS. O LipidEase foi projetado para endereçar as lacunas da área por meio de uma abordagem multifacetada: seu fluxo de trabalho, guiado por uma interface gráfica, abstrai a complexidade dos comandos e minimiza erros operacionais durante a montagem do sistema.

De forma central à proposta, o trabalho aborda diretamente a questão da reprodutibilidade por meio da geração automática de um arquivo de proveniência que registra todos os parâmetros, comandos e versões de software, alinhando-se aos princípios FAIR de gestão de dados [100]. Além do mais, LipidEase introduz uma solução para a portabilidade entre ambientes computacionais, ao gerar pacotes de simulação autocontidos e prontos para execução em clusters de alto desempenho (HPC).

1.1 Objetivos

1.1.1 Objetivo geral

Desenvolver a ferramenta LipidEase para automatizar o preparo e a execução de simulações de dinâmica molecular de bicamadas lipídicas no software GROMACS, incorporando uma interface gráfica de usuário (GUI) que simplifique a configuração e o controle das etapas, bem como mecanismos de padronização e registro automático de proveniência para assegurar a rastreabilidade e a reprodutibilidade dos experimentos computacionais. Além disso, a ferramenta deve permitir a inclusão de fármacos em sistemas lipídicos e gerar pacotes de simulação compatíveis com execução em ambientes de computação de alto desempenho (HPC), promovendo a acessibilidade e a eficiência na condução de estudos envolvendo membranas biológicas.

1.1.2 Objetivos específicos

Para atingir o objetivo geral, estabeleceram-se os seguintes objetivos específicos:

1. Revisar os procedimentos de dinâmica molecular, incluindo as etapas necessárias e os arquivos fundamentais para a execução de simulações envolvendo bicamadas lipídicas.

2. Implementar a automatização da construção do sistema de bicamada lipídica e execução da dinâmica molecular.

3. Implementar uma interface gráfica de usuário (GUI) para controle interativo das etapas de configuração e execução das simulações.

4. Implementar rotinas para geração automática e parametrizada dos arquivos de entrada do GROMACS (.mdp, .top e .itp).

5. Implementar mecanismos para o registro automático dos parâmetros utilizados nas simulações, assegurando a rastreabilidade e a proveniência dos dados gerados.

6. Validar a ferramenta em diferentes configurações de bicamadas lipídicas para verificar sua funcionalidade e reprodutibilidade.

7. Estender a validação da ferramenta a sistemas contendo fármacos, avaliando sua aplicabilidade em simulações de interação fármaco–membrana.

8. Implementar a criação de pacotes autocontidos de simulação compatíveis com execução em ambientes de computação de alto desempenho (HPC).

9. Documentar e disponibilizar o código-fonte da ferramenta em uma plataforma pública, promovendo sua utilização pela comunidade científica. O código pode ser encontrado em: `https://github.com/Paesy/LipidEase`

1.2 Motivação

As membranas celulares exercem funções fundamentais na manutenção e regulação dos processos biológicos, atuando como barreiras seletivas e mediadoras das interações entre o meio intracelular e o ambiente externo. A estrutura dessas membranas é composta por bicamadas lipídicas, cuja organização e propriedades físicas influenciam diretamente fenômenos como permeabilidade, sinalização e transporte de moléculas. Compreender o comportamento dessas bicamadas é essencial para o estudo de processos biológicos e farmacológicos que envolvem a interação entre compostos bioativos e sistemas de membranas.

A dinâmica molecular (DM) tem se mostrado uma das principais abordagens computacionais para investigar sistemas biológicos em escala atômica. Essa técnica permite observar o comportamento temporal de átomos e moléculas, fornecendo informações detalhadas sobre estabilidade estrutural, difusão, flexibilidade e interações intermoleculares. A DM possibilita explorar como diferentes fármacos interagem com membranas celulares, permitindo compreender mecanismos de absorção, permeabilidade e distribuição, etapas cruciais no processo de desenvolvimento de novos medicamentos.

Apesar do avanço das técnicas de simulação, a preparação e execução de dinâmicas moleculares de bicamadas lipídicas ainda exigem múltiplas etapas manuais e a edição direta de arquivos de entrada, o que torna o processo demorado e suscetível a erros. Além disso, a falta de padronização e de registro automático dos parâmetros de simulação compromete a rastreabilidade e a reprodutibilidade dos resultados, dificultando o compartilhamento e a validação por outros pesquisadores. Essas limitações acabam restringindo o uso da dinâmica molecular a usuários com experiência prévia no ambiente de linha de comando e conhecimento técnico específico.

Por conta disso, torna-se necessário o desenvolvimento de uma ferramenta capaz de automatizar e padronizar o fluxo de simulações de bicamadas lipídicas, reduzindo o tempo de configuração, minimizando erros e promovendo a reprodutibilidade científica. A ferramenta proposta, LipidEase, foi concebida com esse propósito: integrar, em um único ambiente, a automação das etapas de preparação e execução das simulações no GROMACS, oferecendo também uma interface gráfica de fácil utilização. A ferramenta permite a inclusão de fármacos nos sistemas lipídicos, ampliando suas possibilidades de aplicação em estudos de interação fármaco–membrana, contribuindo para o avanço de pesquisas na área de bioinformática e no desenvolvimento racional de fármacos.

Assim, a motivação central deste trabalho está na necessidade de criar uma solução que torne a dinâmica molecular mais acessível, reprodutível e eficiente, fortalecendo o uso de métodos computacionais na investigação de sistemas biológicos complexos.

1.3 Organização

O presente trabalho está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica necessária para a compreensão do tema abordado. A Seção 3 traz um breve comparativo entre algumas ferramentas que visam facilitar os processos de dinâmica molecular. A Seção 4 descreve a metodologia empregada no desenvolvimento do estudo. Na Seção 5, são exibidos os resultados obtidos através do uso da ferramenta, assim como seu desenvolvimento. Por fim, a Seção 6 traz as conclusões e sugestões para trabalhos futuros.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta a base teórica fundamental para a compreensão do contexto, das tecnologias e dos desafios relacionados ao desenvolvimento da ferramenta LipidEase. O conteúdo está estruturado de maneira a introduzir progressivamente os principais conceitos que fundamentam a proposta desta pesquisa.

Inicialmente, abordam-se as membranas biológicas e suas características estruturais, com ênfase nas bicamadas lipídicas, por constituírem o objeto central das simulações automatizadas que este trabalho visa facilitar. Em seguida, é apresentada a área de bioinformática, que fornece as bases computacionais para a modelagem e análise de sistemas biológicos complexos.

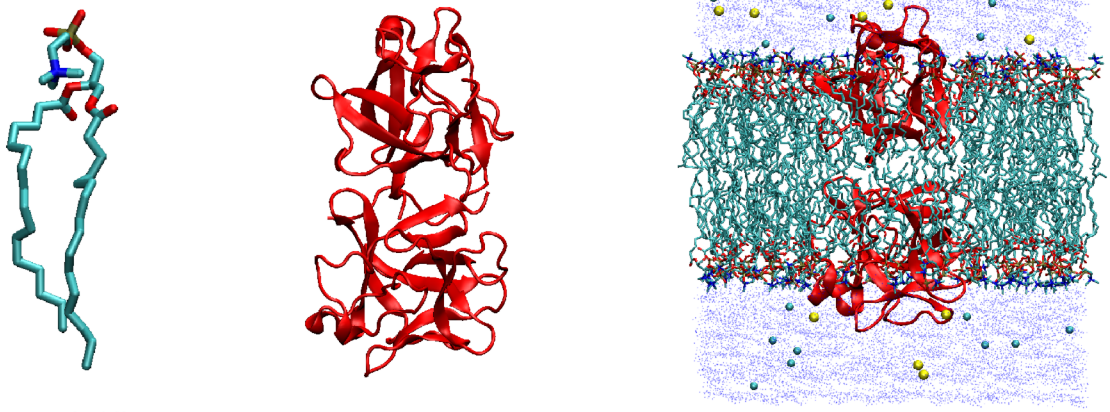
Explora-se a técnica de dinâmica molecular (DM), destacando seus fundamentos teóricos, níveis de simulação, softwares utilizados e sua aplicação na busca e desenvolvimento de fármacos. Também são discutidas as especificidades das simulações envolvendo membranas lipídicas, especialmente no contexto de interação com moléculas bioativas, bem como a importância do GROMACS como ferramenta para a execução dessas simulações.

Complementarmente, este capítulo dedica-se a apresentar o papel da automação de processos científicos, enfatizando os benefícios que essa prática proporciona, como maior eficiência, reprodutibilidade e escalabilidade. Por fim, são discutidos os aspectos relacionados à reprodutibilidade científica, ressaltando sua importância para o avanço da pesquisa computacional e para a validação de resultados em bioinformática e dinâmica molecular.

2.1 Membranas Biológicas

Membranas biológicas são estruturas formadas por lipídios, proteínas e açúcares que interagem de maneira organizada, compondo o entorno de células e organelas. Essa organização garante a separação eficaz entre o meio interno e o meio externo dos organismos, assegurando o isolamento necessário para a manutenção das funções celulares [67]. A matriz estrutural dessas membranas é composta predominantemente por lipídios

[98]. Uma característica central dessas estruturas é sua semipermeabilidade seletiva, que regula a troca de substâncias, garantindo o equilíbrio entre os ambientes. Estudos *in vivo* destacam seu papel no comportamento farmacocinético de substâncias medicamentosas, influenciando processos como difusão, transporte e distribuição molecular no organismo [71]. Essa estrutura confere fluidez, flexibilidade e capacidade de resposta a estímulos ambientais, características indispensáveis à manutenção da homeostase e à regulação de processos biológicos essenciais como sinalização, transporte e comunicação intercelular [74]; um sistema de membrana simples está representado na Figura 1.



(a) Lipídio DPPC

(b) Exemplo de Proteína

(c) Sistema de membrana + proteína inserida, com presença de água e ions

Figura 1: Exemplo da composição de uma membrana biológica.Fonte: Elaborado pelo autor (2025)

2.1.1 Bicamadas Lipídicas

As bicamadas lipídicas constituem a unidade estrutural fundamental das membranas biológicas, sendo formadas pela organização espontânea de moléculas anfipáticas em duas camadas opostas, que possuem uma região polar e hidrofílica, chamada cabeça lipídica, e uma região apolar e hidrofóbica, denominada cauda lipídica. Essa configuração de bicamada resulta da interação entre as regiões das caudas lipídicas, que se afastam do meio aquoso, e as cabeças lipídicas, orientadas para o ambiente externo e interno da célula e permitindo a formação espontânea de uma bicamada lipídica [6, 98], como representado na Figura 2.

Apesar de compartilharem essa arquitetura básica, as bicamadas apresentam uma notável diversidade composicional, determinada pela proporção e pelo tipo de lipídios presentes, como fosfatidilcolina, esfingomielina, colesterol, entre outros. Essa variabilidade confere propriedades físico-químicas distintas às membranas, influenciando aspectos como espessura, fluidez, rigidez e a formação de domínios funcionais, como os *lipid*

rafts [56]. Tais propriedades modulam diretamente a organização e a função de proteínas associadas à membrana, além de impactar processos dinâmicos, como fusão e endocitose.

Em virtude de sua importância biológica e de sua relativa complexidade estrutural [94], as bicamadas lipídicas são amplamente utilizadas como sistemas modelo em estudos experimentais e computacionais [29, 61]. Esses modelos permitem investigar mecanismos moleculares fundamentais, como interações lipídio-proteína, permeabilidade e os efeitos de compostos bioativos, incluindo fármacos e peptídeos [29].

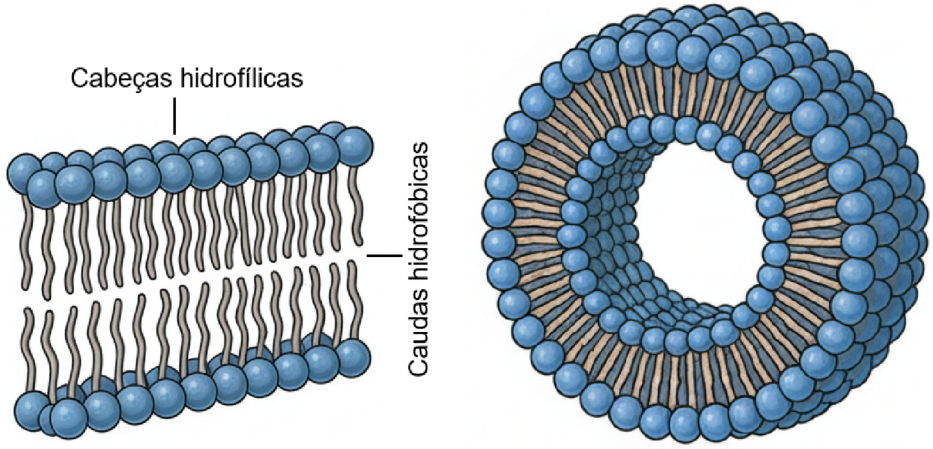


Figura 2: Estruturas de bicamada formadas por moléculas anfipáticas. (A) Bicamada lipídica. (B) Vesícula lipossomal. Fonte: Elaborado pelo autor (2025)

2.1.2 Membranas Biológicas e Fármacos

As membranas biológicas são estruturas fundamentais para a organização e funcionamento das células [22]. Elas delimitam o espaço celular, controlam o transporte de moléculas e participam de processos de reconhecimento e sinalização celular [22]. A bicamada lipídica que as compõe é formada principalmente por fosfolipídios e glicolipídios, cuja composição e fluidez determinam propriedades físico-químicas que são essenciais para a manutenção da integridade celular [77]. Essa complexidade estrutural torna as membranas não apenas componentes passivos, mas também alvos diretos e indiretos de diversas moléculas bioativas [15].

Compreender como fármacos interagem com membranas biológicas é de grande interesse [77]. Em muitos casos, o mecanismo de ação de um composto depende de sua capacidade de atravessar, se ancorar ou modificar a estrutura da bicamada lipídica [62]. Essa relação é particularmente evidente em antibióticos que atuam sobre a parede celular bacteriana, como penicilina, que inibe enzimas responsáveis pela síntese de peptidoglicanos específicos e promovem a desestabilização da membrana [69]. Alterações sutis na composição lipídica — como o comprimento das caudas acilares ou a proporção de fosfa-

tidilgliceróis e cardiolipinas — podem modificar a afinidade e a eficácia dessas moléculas, influenciando diretamente o desenvolvimento de resistência bacteriana.

A modelagem computacional e simulações de dinâmica molecular são ferramentas valiosas para estudar esses processos em escala atômica, permitindo visualizar como pequenas variações estruturais em fármacos ou nos lipídios alteram a organização da bicamada [95]. Esse tipo de abordagem é eficiente para identificar regiões preferenciais de interação, prever mecanismos de inserção ou permeação e apoiar o desenvolvimento racional de compostos mais seletivos e eficazes [30]. Dessa forma, compreender as interações entre fármacos e membranas biológicas é essencial para o avanço da descoberta e otimização de novos agentes terapêuticos [93].

2.2 Bioinformática

A bioinformática é um campo de estudo interdisciplinar que combina ciências computacionais, biológicas, estatística e matemática para interpretação de dados biológicos a partir de um ponto de vista computacional [59]. Esse campo é comumente relacionado à manipulação e análise de sequências de genomas e proteomas, bem como aos dados provenientes de experimentos genômicos e proteômicos. No entanto, também inclui o estudo de modelagem, predição e dinâmica de estruturas macromoleculares, como proteínas, lipídios, carboidratos, DNAs e RNAs [26, 59]. Além da análise de sequências e da predição estrutural, a bioinformática também desempenha um papel crucial na simulação do comportamento dinâmico de macromoléculas biológicas. Entre as diversas metodologias utilizadas nesse contexto, destaca-se a Dinâmica Molecular (DM), uma técnica computacional que permite modelar e compreender, com alto nível de detalhe, a movimentação e as interações entre átomos e moléculas ao longo do tempo.

2.3 Dinâmica Molecular

Dentre as técnicas computacionais amplamente empregadas no âmbito da bioinformática, destaca-se a Dinâmica Molecular considerada uma das abordagens mais versáteis para o estudo de macromoléculas biológicas e o planejamento racional de fármacos baseados em estruturas moleculares [47]. A DM é fundamentada nos princípios da mecânica clássica para calcular as propriedades de equilíbrio, transporte e movimento de um sistema clássico [33].

O sistema dinâmico fornece informações sobre o comportamento atômico, dependente do tempo, de átomos individuais que compõem um sistema, que é resolvido integrando numericamente equações Newtonianas de movimento [37]. Este processo envolve o cálculo das forças que atuam sobre cada átomo em um determinado instante, o deslocamento dos átomos de acordo com essas forças durante um pequeno intervalo de tempo

(Δt), e a repetição desses passos ao longo do tempo desejado para a simulação, como ilustrado esquematicamente na Figura 3.

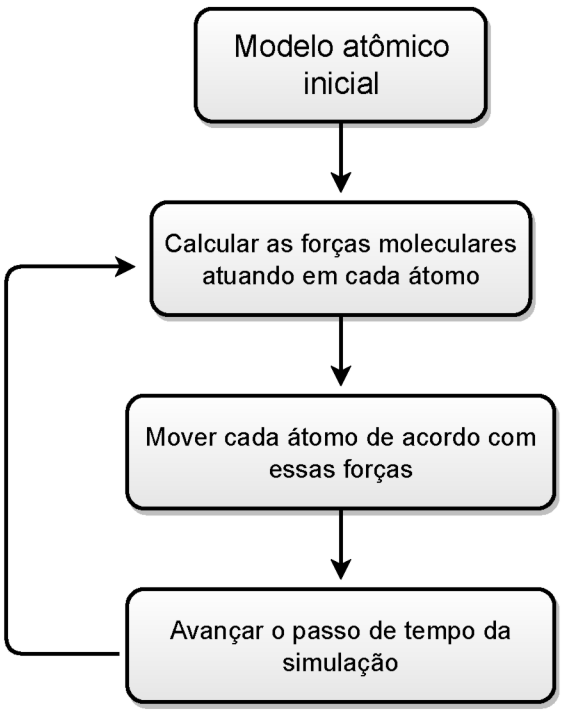


Figura 3: Desenho esquemático da execução de uma dinâmica molecular, ilustrando o ciclo de cálculo de forças e atualização das posições atômicas. Fonte: Elaborado pelo autor (2025)

Sendo assim, simulações por DM predizem como cada átomo em um sistema molecular vai se movimentar durante um tempo determinado. Para calcular as forças que ditam esse movimento, utiliza-se um campo de força: um conjunto de funções matemáticas e parâmetros associados que descrevem a energia potencial do sistema em função das posições atômicas, permitindo assim estimar as interações interatômicas [44, 46]. Ao integrar as leis Newtonianas com as forças derivadas do campo de força, são obtidas sucessivas configurações do sistema simulado, gerando trajetórias que descrevem a velocidade e a posição das partículas ao longo de um período de tempo. A partir dessas trajetórias, podemos calcular várias propriedades do sistema, comparando-as com dados experimentais. Esse processo possibilita uma análise detalhada do comportamento dinâmico das partículas [28, 44].

Para iniciar uma simulação, é necessário dispor de um conjunto inicial de coordenadas atômicas e, opcionalmente, velocidades. Também são essenciais os dados que definem as interações entre os átomos, os quais estão contidos no campo de força. Este é composto porequações (que representam energias de ligação, ângulos, diedros, interações de van der Waals e eletrostáticas) e parâmetros específicos para cada tipo de átomo e ligação (como constantes de mola, cargas parciais, raios atômicos). É importante notar que esses parâmetros são geralmente derivados de cálculos de mecânica quântica ou

ajustados para reproduzir dados experimentais de moléculas pequenas ou propriedades macroscópicas [44, 47]. Separadamente, as coordenadas atômicas iniciais, que definem a estrutura tridimensional de partida para a simulação, são frequentemente obtidas por meio de técnicas experimentais como cristalografia de raio-X, ressonância magnética nuclear ou crio-microscopia eletrônica, ou ainda através de modelagem computacional [47]. O sistema modelado, a partir dessas coordenadas iniciais, requer processos preliminares como minimização de energia e equilíbrio térmico e de pressão, a fim de alcançar uma configuração estável para a simulação. A não execução desses procedimentos pode comprometer a estabilidade do sistema e gerar resultados errôneos.

2.3.1 Interações moleculares e equações fundamentais em Dinâmica Molecular

O cálculo das forças que atuam sobre cada átomo em uma simulação de Dinâmica Molecular (DM) deriva da energia potencial total do sistema. Em campos de força clássicos, essa energia é representada como a soma de termos ligados (ligações, ângulos e diedros) e não ligados (interações de van der Waals e eletrostáticas) [12, 32, 33, 38].

De forma geral, a energia total pode ser expressa como:

$$E_{\text{total}} = E_{\text{lig}} + E_{\text{ang}} + E_{\text{diedro}} + E_{\text{LJ}} + E_{\text{Coulomb}}. \tag{1}$$

2.3.1.1 Potencial de Lennard-Jones

As interações de van der Waals são comumente descritas pelo potencial de Lennard-Jones, que modela a repulsão em curtas distâncias e a atração devida a forças de dispersão:

$$V_{\text{LJ}}(r) = 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right]. \tag{2}$$

Esse termo influencia diretamente o empacotamento lipídico, a fluidez e as propriedades mecânicas de bicamadas [30, 75].

2.3.1.2 Interações eletrostáticas

As interações coulômbicas entre cargas parciais são descritas pela forma clássica da Lei de Coulomb: w

$$V_{\text{Coulomb}}(r) = \frac{1}{4\pi\epsilon_0}\frac{q_i q_j}{r}. \tag{3}$$

Essas interações são essenciais para descrever adequadamente a interface água–membrana, a organização das cabeças polares e o papel de íons multivalentes [62, 95].

2.3.1.3 Integração das equações de movimento

As forças derivadas dos potenciais alimentam as equações de Newton:

$$m_i \frac{d^2\mathbf{r}_i}{dt^2} = -\nabla_i E_{\text{total}}, \tag{4}$$

que são resolvidas numericamente por algoritmos de integração de passo finito, como *leap-frog* ou *velocity-Verlet*, amplamente empregados em simulações biomoleculares [32, 37, 46].

2.3.2 Condições periódicas de contorno (Periodic Boundary Conditions – PBC)

Simulações de DM utilizam condições periódicas de contorno (PBC) para evitar efeitos artificiais decorrentes das bordas finitas da caixa de simulação. Nessa abordagem, o sistema é replicado infinitamente em todas as direções, de modo que, quando uma molécula cruza uma fronteira, ela reaparece no lado oposto. Isso permite aproximar o comportamento de um sistema volumétrico a partir de uma amostra relativamente pequena [12, 32, 38].

Em sistemas de membranas lipídicas, o uso de PBC é particularmente importante porque:

- elimina efeitos de superfície que distorceriam a interface água–membrana;
- permite difusão contínua no plano da bicamada;
- evita a exposição artificial das caudas lipídicas ao vácuo;
- auxilia na manutenção de uma densidade lipídica realista ao longo da simulação.

Diretrizes específicas para boas práticas de simulação de membranas destacam explicitamente a necessidade de PBC e de caixas com dimensões adequadas para minimizar correlações artificiais entre as imagens periódicas [30, 85].

2.3.3 Níveis de Simulação por Dinâmica Molecular

2.3.3.1 All-atom

O modelo *all-atom* (AA) representa o sistema simulado em seu nível mais detalhado, onde cada átomo no sistema, seja soluto ou solvente, é explicitamente modelado. Isso significa que todos os átomos e seus parâmetros (*e.g* massa, carga) são considerados individualmente no sistema [12]. Devido à representação individual dos átomos e aos recursos computacionais disponíveis, as simulações são limitadas em relação à escala de tempo (Fig. 4) utilizada em nanosegundos, que acaba limitando quais eventos biológicos somos capazes de observar através da dinâmica molecular [106].

2.3.3.2 Coarse-Grained

Simulações *coarse-grained* (CG) têm sido extensivamente utilizadas para simular processos moleculares complexos com baixo custo computacional em relação às simulações AA [39]. Contudo, mesmo que os modelos CG possibilitem simulações mais rápidas e por intervalos de tempo maiores, sua resolução do sistema é menor. Isso ocorre pois as

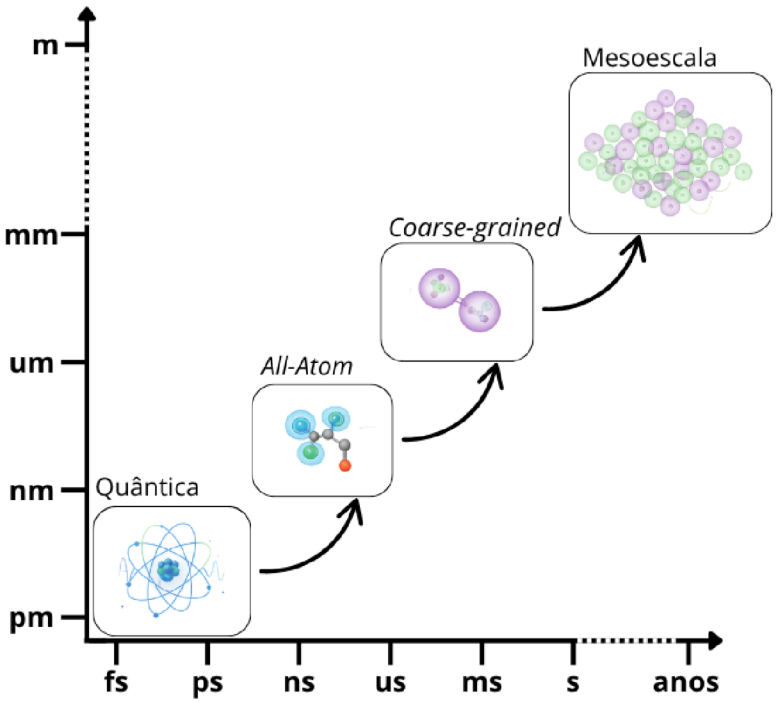


Figura 4: Escala de tempo aproximada de modelagem molecular nos diferentes tamanhos dos sitemas: Quântico, *all-atom*, *coarse-grained* e mesoescala. Fonte: Elaborado pelo autor (2025)

simulações CG abordam o sistema atômico de maneira simplificada, agrupando grupos de átomos em *beads*, reduzindo os graus de liberdade presentes nos átomos e interações atômicas computadas [41].

Há várias desvantagens associadas às simulações CG, sendo os principais desafios a imprecisão, a falta de transferibilidade e a omissão das interações não nativas dos átomos. Os esforços para desenvolver modelos CG em diferentes escalas têm contribuído para mitigar, mas não necessariamente eliminar, os dois problemas mencionados anteriormente. [90].

Ainda assim, as simulações CG apresentam vantagens relevantes, especialmente em sistemas de membranas biológicas, pois permitem aumentar significativamente a escala temporal e espacial acessível [87], reduzir o custo computacional e capturar processos de auto-organização e fusão de bicamadas que seriam inviáveis em simulações all-atom. Essa abordagem é particularmente útil para investigar fenômenos de larga escala, como a formação de domínios lipídicos ou a inserção de proteínas na membrana [13, 80].

2.3.3.3 Quântica

As dinâmicas moleculares quânticas (DMQ) são realizadas através de fundamentos da mecânica quântica. Essas teorias descrevem o comportamento das partículas em nível atômico e subatômico. Ao contrário dos outros métodos mencionados anteriormente, a DMQ considera explicitamente o comportamento de onda das partículas, incorporando

princípios como a dualidade onda-partícula e o princípio da incerteza de Heisenberg, que estabelece que não é possível determinar simultaneamente, com precisão, a posição e o momento linear de uma partícula [72]. A equação de Schrödinger é utilizada para determinar o estado quântico do sistema durante a produção por dinâmica molecular [21, 101], demandando uso extensivo de recursos computacionais para resolver a equação para sistemas complexos. Algumas aproximações numéricas podem ser aplicadas para tornar os cálculos da dinâmica viáveis computacionalmente, como a implementação dos métodos da teoria do funcional da densidade (DFT) [45, 96].

Este método é muito utilizado para a compreensão de reações químicas nos sistemas [83], principalmente no detalhamento de interações de catalização enzimáticas e estados de protonação.

2.3.4 Softwares para Dinâmica Molecular

Diversas ferramentas computacionais são amplamente utilizadas para executar processos de dinâmica molecular, cada uma com características que as tornam adequadas para diferentes tipos de simulação. Dentre elas, destacam-se pacotes como *GROMACS* [5], AMBER [19], NAMD [73], CHARMM [14] e LAMMPS [91]. Essas ferramentas oferecem suporte para simulações avançadas de moléculas bioquímicas, variando em especializações, eficiência computacional e aplicações específicas.

Além dos pacotes de simulação, existem ferramentas como PyMol [25] e VMD (*Visual Molecular Dynamics*) [40], empregadas para inspeção visual e análise das trajetórias geradas pelas dinâmicas moleculares, facilitando a interpretação e a apresentação dos resultados.

2.3.4.1 GROMACS

GROMACS (*GROningen MAchine for Chemical Simulations*) [5] é um pacote versátil de simulações por DM, gratuito e *open-source*. O software foi projetado para resolver as equações de movimento newtonianas em sistemas contendo milhares ou até milhões de partículas. Seu alto desempenho se deve à implementação otimizada de algoritmos paralelos, que exploram recursos de múltiplos núcleos de processadores e unidades de processamento gráfico (GPU), permitindo simulações em larga escala com precisão temporal e espacial adequadas. Essa característica o torna uma das escolhas mais populares para a modelagem de biomoléculas, como proteínas, lipídios e ácidos nucleicos. Apesar de contar com uma documentação e comunidade de usuários extensas, o programa apresenta uma curva de aprendizado acentuada [81]. Além disso, a ausência de interface gráfica dificulta seu uso por usuários com pouca experiência na área de DM e execução de linha de comando.

2.3.4.2 AMBER

O pacote AMBER (*Assisted Model Building with Energy Refinement*) é amplamente utilizado para simulações de biomoléculas, como proteínas e ácidos nucléicos. Ele é licenciado pela University of California, San Francisco para diferentes tipos de uso — para instituições académicas e sem fins lucrativos costuma haver licença institucional; já o uso comercial implica aquisição de licença específica [19]. Em contrapartida, a suíte *AmberTools*, que oferece ferramentas de preparação de sistemas, análise de trajetórias e parametrizações auxiliares, é disponibilizada gratuitamente e sob licenças *open-source* (por exemplo, Licença Pública Geral (GPL)) [18]. Essa distinção amplia a aplicabilidade do AMBER no ambiente académico, pois pesquisadores podem usar AmberTools livremente e, caso disponham da licença institucional ou de aquisição, utilizar o AMBER completo.

2.3.4.3 CHARMM

CHARMM (*Chemistry at HARvard Macromolecular Mechanics*) [14] é um dos pacotes mais antigos e amplamente utilizados para simulações moleculares. Desenvolvido inicialmente para o estudo de macromoléculas biológicas, como proteínas e DNA, o CHARMM é referência devido à sua abordagem robusta para cálculos de interação molecular. O programa também oferece flexibilidade para modelar sistemas não convencionais, como materiais e interfaces. Seus campos de força detalhados são utilizados tanto na parametrização de sistemas moleculares quanto como base para outros pacotes de simulação.

2.3.4.4 NAMD

NAMD (*Nanoscale Molecular Dynamics*) [73] é um *software* de simulação molecular conhecido por sua eficiência e escalabilidade em sistemas de grande porte. Otimizado para clusters e supercomputadores, ele permite simulações paralelas de alta performance em sistemas contendo milhões de átomos. NAMD é amplamente utilizado em estudos de sistemas complexos, como grandes membranas biológicas, e é frequentemente integrado ao VMD para análise e visualização de simulações. Para usuários mais experientes, possui mais flexibilidade e customização de parâmetros, tornando-o uma escolha popular em projetos computacionalmente intensivos.

2.3.4.5 LAMMPS

LAMMPS (*Large-scale Atomic/Molecular Massively Parallel Simulator*) [91] é um pacote voltado para simulações em larga escala, amplamente usado para estudar materiais e sistemas heterogêneos. Sua arquitetura modular permite modelar uma variedade de interações, desde materiais sólidos até fluidos complexos, além de incluir suporte para simulações de biomoléculas. Por ser gratuito e *open-source*, LAMMPS é ideal para pes-

quisadores que desejam adaptar o software para atender a necessidades específicas de simulação molecular ou de materiais.

2.3.5 Comparação entre os pacotes de simulação molecular

A Tabela 1 apresenta uma análise comparativa entre alguns dos principais programas utilizados em simulações por DM, considerando suas funcionalidades gerais. Foram avaliados critérios como o suporte a aceleração por unidades de processamento gráfico (GPU), a capacidade de realizar minimização de energia (Mini), a execução de DM, a possibilidade de modelagem molecular (Mod) e a existência de recursos de visualização tridimensional (Vis 3D).

De modo geral, todos os programas listados suportam minimização de energia e execução de dinâmicas moleculares, além de oferecerem compatibilidade com GPU, o que gera ganhos de desempenho em simulações de larga escala. Contudo, apenas alguns deles apresentam recursos internos para modelagem de moléculas ou construção de sistemas, como o *AMBER*, o *CHARMM* e o *LAMMPS*, enquanto outros, como o *GROMACS* e o *NAMD*, dependem de ferramentas auxiliares (por exemplo, o *CHARMM-GUI* ou o *VMD*) para a preparação das estruturas. Nenhum dos pacotes analisados possui, de forma nativa, um ambiente completo de visualização tridimensional, sendo essa função geralmente desempenhada por softwares externos especializados, como *VMD*, *PyMOL* ou *Chimera*.

A Tabela 2 complementa essa análise, destacando as abordagens de simulação e os tipos de sistemas moleculares suportados por cada programa. Nela, são comparadas as capacidades de tratamento de modelos *coarse-grained* (CG) e *all-atom* (AA), o suporte a métodos híbridos quânticos-moleculares (QM/MM), e as aplicações mais comuns em proteínas, membranas e sistemas envolvendo fármacos.

Observa-se que o *GROMACS*, o *AMBER*, o *CHARMM* e o *NAMD* apresentam amplo uso de moléculas, incluindo proteínas e membranas, além de oferecerem integração com metodologias de parametrização de ligantes. O *LAMMPS*, por sua vez, destaca-se pela flexibilidade e abrangência, sendo capaz de realizar simulações atomísticas, *coarse-grained* e quânticas, embora exija maior esforço de configuração e conhecimento de scripts para uso.

O software *GROMACS* foi escolhido neste trabalho por apresentar alto desempenho computacional, ampla compatibilidade com o campo de força *CHARMM* e suas variações (incluindo o CGenFF), além de permitir automação completa das etapas de simulação via linha de comando. Em comparação, o *NAMD*, embora também eficiente e amplamente utilizado, depende de scripts externos e de ferramentas auxiliares como o *VMD* para etapas de preparação e análise, o que torna sua integração em fluxos automatizados menos direta.

Nota: O símbolo **X** indica suporte nativo e consolidado da funcionalidade em questão,

enquanto **(X)** representa suporte parcial ou dependente de módulos, extensões ou programas auxiliares. O símbolo **–** indica ausência ou uso limitado da funcionalidade.

Tabela 1: Comparação entre os principais programas de dinâmica molecular.

Programa	GPU	Mini	DM	Vis 3D	Mod
GROMACS	X	X	X	–	(X)
AMBER	X	X	X	–	X
CHARMM	X	X	X	–	X
NAMD	X	X	X	–	(X)
LAMMPS	X	X	X	(X)	(X)

Tabela 2: Comparação entre os principais programas de dinâmica molecular.

Programa	CG	AA	Quântica	Proteínas	Membranas	Fármacos
GROMACS	X	X	(X)	X	X	X
AMBER	(X)	X	(X)	X	X	X
CHARMM	(X)	X	X	X	X	X
NAMD	(X)	X	(X)	X	X	X
LAMMPS	X	X	X	(X)	(X)	(X)

2.3.6 Dinâmica Molecular e Busca de Fármacos

A busca de novos fármacos por meio de métodos computacionais é uma estratégia essencial para reduzir o tempo e o custo associados ao desenvolvimento farmacêutico, além de aumentar a precisão na seleção de candidatos promissores [44, 79]. Nesse contexto, a DM permite explorar, em escala atômica, o comportamento estrutural e funcional de moléculas e seus complexos em ambientes simulados que reproduzem condições fisiológicas [77].

Diferentemente de abordagens estáticas, como a modelagem molecular ou o *docking*, as simulações por DM consideram a flexibilidade de moléculas, possibilitando observar flutuações conformacionais, rearranjos de sítio de ligação e interações dinâmicas entre ligantes e receptores. Essa capacidade é crucial, uma vez que a afinidade e a estabilidade de um fármaco dependem não apenas da complementaridade geométrica, mas também das flutuações energéticas e estruturais que ocorrem durante o processo de ligação [37].

O processo computacional de descoberta de fármacos normalmente se inicia com a identificação e validação de alvos biológicos relevantes, como proteínas, receptores, enzimas ou ácidos nucleicos, que possam ser modulados para prevenir ou tratar doenças [79]. A dinâmica molecular contribui nessa etapa ao permitir o refinamento de estruturas obtidas experimentalmente (por exemplo, via cristalografia de raios X ou criomicroscopia eletrônica) e ao revelar estados conformacionais alternativos que podem representar alvos mais acessíveis para a ligação de ligantes [28].

As simulações de DM também são amplamente empregadas para avaliar a estabilidade e o comportamento de complexos proteína–ligante, permitindo acompanhar o processo de ligação e desligamento [28], identificar interações hidrofóbicas e eletrostáticas relevantes e compreender a contribuição dinâmica das moléculas de água e íons na estabilidade do sistema. Essas informações auxiliam na priorização de compostos com maior potencial terapêutico antes de etapas experimentais mais custosas [34].

Em sistemas biológicos mais complexos, como membranas lipídicas, a DM permite estudar processos fundamentais relacionados à entrega de fármacos, incluindo difusão lateral, permeabilidade, partição de compostos hidrofóbicos e efeitos de ancoragem molecular. Dessa forma, a simulação de interações entre fármacos e membranas fornece percepções sobre mecanismos de absorção, estabilidade e bioacessibilidade em nível molecular, aspectos críticos para a compreensão da eficácia de moléculas candidatas a fármacos [60, 77].

Assim, a flexibilidade e o detalhamento temporal oferecidos pelas simulações de dinâmica molecular tornam essa técnica indispensável para compreender, prever e otimizar a interação entre compostos farmacológicos e seus alvos biológicos, integrando-se de maneira estratégica aos fluxos modernos de *drug discovery* e *drug design*.

2.3.6.1 Simulações Membrana x Fármaco

O contexto da ação de fármacos por DM é capaz de elucidar informações que não são possíveis de responder no paradigma da “chave-fechadura” , especialmente no que tange à dinâmica e ao ambiente complexo das membranas lipídicas [77]. Enquanto o modelo tradicional foca na interação estática com um alvo proteico isolado, a realidade biológica exige a compreensão de como o fármaco se comporta e interage dentro e através das biomembranas. As biomembranas são estruturas fluidas complexas formadas principalmente por bicamadas de fosfolipídios, podendo ser consideradas os blocos de construção primários da vida [58, 68, 98]. Devido à importância das biomembranas na organização celular, torna-se essencial entender o contexto de interação dos fármacos com este componente fundamental. O acesso do fármaco ao seu alvo envolve interação com biomembranas, seja atravessando a biomembrana ou interagindo com a proteína-alvo, que muitas vezes está associada à membrana. Há casos em que o alvo do fármaco é a própria membrana, e não uma proteína específica, como alguns agentes antimicrobianos ou antitumorais que desestabilizam a integridade da bicamada lipídica [77].

Alguns trabalhos demonstraram como a DM colaborou para elucidar o papel que a membrana lipídica exerce no substrato e, consequentemente, na seleção de fármacos para proteínas de membrana [79, 104, 105], interações membrana-fármaco [57, 62], liberação de fármacos [48, 53], peptídeos antimicrobiais [24, 66] e metodologias [55].

As interações entre fármacos e membranas lipídicas representam uma etapa crítica na farmacocinética e farmacodinâmica dos fármacos, determinando propriedades como

absorção, distribuição e metabolismo. As simulações de DM são ferramentas amplamente utilizadas para investigar esses processos em nível molecular, permitindo observar como moléculas de fármacos se organizam, difundem e se estabilizam na interface lipídica [57, 62]. Por meio dessas simulações, é possível analisar a penetração de compostos hidrofóbicos nas regiões apolares da bicamada, a formação de agregados moleculares, a perturbação da estrutura lipídica e os efeitos de ancoragem promovidos por grupos funcionais específicos. Tais análises ajudam a compreender como as propriedades físico-químicas do fármaco influenciam sua afinidade pela membrana. Os resultados obtidos a partir dessas simulações fornecem informações complementares para o desenho racional de moléculas com maior biodisponibilidade e seletividade, especialmente no desenvolvimento de fármacos direcionados a proteínas de membrana.

2.3.7 DM de Lipídios - *GROMACS*

Para a execução do processo de dinâmica molecular (DM) de lipídios utilizando o *GROMACS*, são necessários arquivos fundamentais para a construção do sistema. Esses arquivos incluem informações estruturais das moléculas (como coordenadas atômicas), parâmetros do campo de força (que definem as interações entre os átomos), arquivos de topologia que descrevem detalhadamente as propriedades e conectividades das moléculas e arquivos específicos para os processos de minimização de energia e equilíbrio do sistema simulado.

A construção e simulação de uma bicamada lipídica, especialmente no contexto de interações com fármacos, envolve uma sequência de etapas que demandam atenção meticulosa e conhecimento dos comandos do *GROMACS*. A seguir, são descritos os passos típicos para realizar uma simulação de DM de lipídios, que serão revisitados na Seção 5.4, onde cada etapa é discutida e ilustrada com exemplos visuais.

2.3.7.1 Preparação da Estrutura Inicial

Embora o *GROMACS* possua a alternativa `pdb2gmx`, que gera automaticamente arquivos de topologia a partir de estruturas moleculares, ela é principalmente voltada para proteínas e ácidos nucleicos, que já possuem parâmetros bem definidos nos campos de força usados nesse programa. No caso dos lipídios, muitos não possuem parâmetros completos nos campos de força padrão do *GROMACS*. Além disso, as membranas são sistemas muito complexos e sensíveis, e por isso é importante utilizar campos de força bem validados, específicos para esses sistemas. Por esse motivo, é mais seguro e comum utilizar ferramentas externas, como o CHARMM-GUI [102], que já possuem bibliotecas prontas de lipídios e geram arquivos de topologia compatíveis com o *GROMACS*, e CGenFF web-server, que gera parâmetros de campo de força para moléculas que ainda não estão definidas na biblioteca principal de campos de força CHARMM. Assim, garantimos que a simulação será mais precisa e fiel ao comportamento real das membranas.

Após selecionar o tipo de lipídio e sua proporção (quantidade de lipídios na membrana), a construção da bicamada no *GROMACS* exige uma série de passos manuais. É necessário utilizar comandos como `editconf` para alinhar o lipídio no eixo normal (geralmente o eixo Z) e delimitar os limites (eixos) da caixa de simulação no entorno desse lipídio. Em seguida, o comando `genconf` é empregado para construir uma monocamada através da replicação do lipídio inicial.

2.3.7.2 Construção da Bicamada

Para formar a bicamada, a primeira monocamada criada precisa ser invertida e deslocada no eixo Z usando novamente o comando `editconf`, o que é crucial para evitar colisões atômicas quando as duas estruturas forem unidas. As duas monocamadas (original e invertida/deslocada) são então concatenadas utilizando o comando cat (por exemplo, `cat` monocamada.gro inversa.gro $>$ bicamada.gro). Este processo, entretanto, não é trivial, pois exige edições manuais subsequentes para ajustar a quantidade de átomos no novo arquivo .gro gerado, garantir a formatação correta e, fundamentalmente, atualizar a quantidade de átomos e moléculas no arquivo de topologia (.top) do sistema. Esta fase de construção manual é suscetível a erros, reforçando a necessidade de métodos automatizados.

2.3.7.3 Solvatação e Neutralização de Cargas

A etapa de solvatação é crucial para simular o ambiente biológico da bicamada lipídica. Para adicionar moléculas de água ao sistema no *GROMACS*, emprega-se uma abordagem que envolve primeiramente a solvatação de uma caixa de simulação de água separada. Isso é feito utilizando o comando `gmx solvate` e o tipo de água escolhido pelo usuário de acordo com a literatura (como TIP3P, TIP4P ou SPC), com dimensões que se adequem ao tamanho do sistema lipídico já construído.

Posteriormente, essa caixa solvatada é duplicada, e suas cópias são inseridas nas porções inferior e superior da bicamada lipídica, deixando ao menos 0,1 nm de distância entre as caixas de água e as cabeças polares dos lipídios. Essa inserção é feita através de deslocamentos precisos no eixo Z das caixas de água, assegurando que não haja colisões com a membrana lipídica. Após o posicionamento, os arquivos são concatenados, e é necessária a atualização manual do arquivo de topologia para a adição das moléculas de solvente, assim como o tratamento do arquivo do sistema solvatado. Essa atenção aos detalhes e as intervenções manuais sublinham a complexidade da preparação de um sistema para a dinâmica molecular.

Caso o sistema tenha moléculas adicionais de estudo, como potenciais fármacos, é criada uma caixa com essas moléculas, e essa caixa é solvatada e depois adicionada ao sistema, seguindo os mesmos passos de atualização e concatenação de arquivos.

Com o sistema solvatado, o próximo passo é a adição de íons (como Na^+ e Cl^-) para

neutralizar a carga total do sistema e ajustar a força iônica. Este ajuste é fundamental para replicar o ambiente fisiológico e garantir a estabilidade elétrica da simulação.

2.3.7.4 Execução dos Processos de Dinâmica

Após a construção inicial do sistema, as simulações de DM exigem etapas de minimização de energia e equilíbrio (NVT e NPT). A minimização de energia é fundamental para remover interações de alta energia que podem surgir da construção inicial do sistema, garantindo o relaxamento das moléculas. Em seguida, as etapas de equilíbrio, como o NVT e o NPT, são executadas por um período reduzido e são necessárias para atingir a organização adequada da bicamada e das moléculas do sistema. Durante essa fase, aplicações de restrição de movimento dos átomos são adicionadas para que o sistema atinja o equilíbrio de forma controlada.

2.3.7.5 Dinâmica e Geração de Trajetória

Por fim, executa-se outra etapa da fase NPT, porém sem restrições de movimento e por período prolongado (nanossegundos a microssegundos). É a partir deste procedimento que se torna possível observar fenômenos de interesse biológico, como a interação fármaco-membrana, permitindo a coleta de dados dinâmicos essenciais para a compreensão do comportamento molecular.

Como foi evidenciado nos passos descritos nesta seção, a execução de uma DM de bicamadas lipídicas envolve uma série de etapas complexas e interdependentes, que demandam atenção e conhecimento especializado. Cada fase requer a escolha e ajuste de diversos parâmetros que influenciam diretamente a qualidade e a reprodutibilidade dos resultados obtidos. Assim, torna-se evidente a importância de automatizar esses procedimentos, tanto para reduzir a possibilidade de erros humanos quanto para padronizar a execução das simulações. Além disso, o registro dos parâmetros utilizados é essencial para obter-se a transparência, rastreabilidade e a reprodutibilidade científica. Assim, fica em evidência a necessidade de ferramentas computacionais que integrem e automatizem esses processos.

2.4 Automação de Processos Científicos

A automação de processos científicos refere-se ao uso de ferramentas computacionais para executar tarefas repetitivas e complexas de forma eficiente e padronizada [36]. No contexto da bioinformática e DM, a automação desempenha um papel fundamental na reprodutibilidade dos experimentos e na redução do tempo gasto em procedimentos operacionais. A crescente complexidade das análises e a necessidade de garantir a integridade dos dados tornam a automação uma estratégia essencial para aumentar a confiabilidade e a escalabilidade dos experimentos computacionais. A automação tem-se tornado essen-

cial na ciência computacional, especialmente em campos como bioinformática e dinâmica molecular. Sua implementação permite maior reprodutibilidade, eficiência e precisão nos experimentos, facilitando o desenvolvimento de novas metodologias e avanços científicos.

2.4.1 Benefícios da Automação na Bioinformática e Dinâmica Molecular

A aplicação da automação na bioinformática e na DM traz benefícios significativos, especialmente no contexto atual de ciência orientada por dados. No artigo publicado Amaro et al. (2025) [8], o armazenamento estruturado e padronizado das trajetórias de simulações amplia consideravelmente o potencial de reuso dos dados, possibilitando novas análises, meta-análises e validações cruzadas que antes seriam inviáveis devido à falta de acessibilidade e interoperabilidade. Esse cenário beneficia diretamente o desenvolvimento de modelos preditivos mais robustos, a otimização de campos de força e a aceleração de descobertas científicas. Além disso, a automação facilita a integração desses dados com bancos públicos e iniciativas colaborativas, fomentando uma ciência mais aberta e colaborativa, alinhada aos princípios da sustentabilidade e da eficiência no uso de recursos computacionais. A implementação de scripts e *workflows* automatizados proporciona benefícios como:

- Maior eficácia e produtividade: as tarefas repetitivas de preparo de sistemas, considerando os mais variados parâmetros a serem selecionados, podem ser realizadas automaticamente.
- Maior reprodutibilidade: os processos automatizados reduzem a variabilidade dos experimentos, garantindo maior padronização dos estudos.
- Escalabilidade: a automatização permite larga escala de execução dos experimentos sem a necessidade de interferência humana frequente.
- Redução de erros humanos: ao minimizar a intervenção manual, a automação diminui a ocorrência de erros operacionais, garantindo a configuração correta dos parâmetros.

2.5 Reprodutibilidade Científica

As dinâmicas moleculares tornaram-se mais acessíveis e poderosas nos últimos anos, impulsionadas por avanços significativos no hardware de computadores convencionais e clusters, além do desenvolvimento de interfaces mais intuitivas para a preparação e execução de simulações [31, 88]. Esses avanços promovem a eficiência das dinâmicas, assim como a melhoria na reprodutibilidade científica das execuções de dinâmicas moleculares, pois o desenvolvimento de novas ferramentas oferece maior suporte à padronização de protocolos, permitindo o compartilhamento de parâmetros de simulação para replicação por diferentes grupos de pesquisa [4, 11].

Além disso, o uso de *workflows* e scripts tem possibilitado a documentação detalhada das etapas do processo, desde a preparação do sistema até a análise dos resultados. Esses scripts podem ser compartilhados em repositórios públicos, promovendo a colaboração entre pesquisadores e transparência na execução de dinâmicas moleculares [9, 42, 50]. Esses avanços, tanto tecnológicos quanto metodológicos, têm reduzido incoerências na reprodutibilidade dos estudos computacionais voltados para dinâmica molecular, tornando a área mais robusta e confiável [3, 17, 49].

A complexidade e escala das simulações de dinâmica molecular, promovidas pelos avanços computacionais, geram um volume massivo de dados que frequentemente são subutilizados ou descartados após a publicação dos resultados. Conforme argumentado por Amaro et al. (2025) [8], há uma necessidade urgente de transformar esse paradigma, adotando princípios FAIR (*Findable, Accessible, Interoperable, Reusable* - Encontrável, Acessível, Interoperável, Reutilizável) para garantir que os dados gerados sejam adequadamente arquivados e possam ser reutilizados por outras pesquisas. Esse movimento não apenas favorece a reprodutibilidade científica, mas também impulsiona a automação de processos, ao permitir a integração sistemática de dados e resultados em fluxos de trabalho computacionais otimizados, promovendo eficiência e sustentabilidade no campo da bioinformática e da dinâmica molecular.

No entanto, a reprodutibilidade em simulações de DM, especialmente no nível binário, é complexa devido a erros numéricos. Isso dificulta a reprodução exata da mesma trajetória, mesmo que se utilizem os mesmos parâmetros e arquivos. Além disso, a construção manual de sistemas pode gerar divergências ainda maiores nos resultados.

A integração entre automação e reprodutibilidade constituem os princípios sobre os quais o desenvolvimento da ferramenta proposta nesta dissertação é baseado. Ao automatizar etapas de preparação e execução de simulações de dinâmica molecular, procura-se reduzir a necessidade de intervenção manual, padronizar os procedimentos de execução e garantir a consistência entre diferentes usos da ferramenta. Essa abordagem melhora a eficiência computacional e o aproveitamento de recursos, assim como auxilia a reprodutibilidade científica, aspectos essenciais na bioinformática moderna. A ferramenta desenvolvida representa um esforço para alinhar as práticas de DM aos princípios FAIR e com isso, promover um ambiente mais confiável e colaborativo para estudos de DM.

3 TRABALHOS RELACIONADOS

A automação de simulações por dinâmica molecular tem sido uma área de crescente interesse e desafio na biologia computacional, e inúmeros esforços são realizados para automatizar esses processos. Diversas ferramentas têm sido desenvolvidas para facilitar a execução e análise dessas simulações, minimizando a complexidade do processo e permitindo maior reprodutibilidade dos experimentos. No entanto, muitas dessas abordagens apresentam limitações, como a falta de flexibilidade na configuração do sistema, dependência de interfaces gráficas, dificuldades na integração com fluxos de trabalho personalizados, limitações dos sistemas de estudo ou falta de portabilidade das ferramentas.

Neste capítulo, são discutidos trabalhos que buscaram otimizar a execução de dinâmicas moleculares, destacando suas principais contribuições, limitações e a motivação para o desenvolvimento da ferramenta proposta. O objetivo é contextualizar a necessidade de uma solução mais acessível e eficiente para a automatização de simulações de bicamadas lipídicas.

3.1 MDCrow

O estudo de Campbell et al. (2025) [16] apresenta o *MDCrow*, um assistente para dinâmicas moleculares baseado em modelos de linguagem de larga escala (LLMs), projetado para automatizar tarefas complexas em simulações de DM. O *MDCrow* utiliza uma abordagem denominada *chain-of-thought*, na qual o modelo realiza um raciocínio interno passo a passo para planejar e decidir qual ferramenta empregar e quais parâmetros aplicar em cada etapa da simulação. Essa cadeia de raciocínio não é exibida ao usuário, mas permite que o sistema organize suas ações, reconhecendo, por exemplo, que é necessário preparar e corrigir arquivos de estrutura, solvatar o sistema e definir o campo de força antes de iniciar a execução de uma simulação.

A ferramenta integra mais de quarenta ferramentas especializadas que abrangem todas as etapas do fluxo de trabalho de uma simulação, desde a preparação até a análise. Entre eles estão rotinas para manipulação e limpeza de arquivos *PDB*, definição de topologias e parâmetros, execução de simulações com o *OpenMM* e análise de trajetórias por meio

de bibliotecas como o *MDTraj*. A interação com o *MDCrow* ocorre através de um chat: o usuário descreve a tarefa desejada, e o sistema interpreta o pedido, gera internamente o plano de execução por meio do raciocínio *chain-of-thought*, cria os comandos correspondentes em *Python* e os executa automaticamente. Todo o processo, incluindo arquivos de entrada, resultados e gráficos, é organizado em diretórios específicos, o que permite retomar a simulação ou prosseguir com análises adicionais.

Os autores avaliaram o desempenho do *MDCrow* em 25 tarefas distintas, demonstrando que modelos como o *GPT-4o* e o *Llama3-405b* alcançaram altas taxas de sucesso na execução de processos complexos. O estudo também observou que o estilo de comando influencia significativamente o desempenho de modelos menores, mas apresenta pouco impacto em modelos mais robustos. O *MDCrow* destaca-se por sua capacidade adaptativa, sendo capaz de ajustar seus fluxos de trabalho a diferentes ferramentas e condições de simulação, sem depender de configurações rígidas específicas para cada software.

Esse trabalho se alinha a iniciativas anteriores de automação em DM, como o *PyAutoFEP* — uma ferramenta desenvolvida em Python3 que automatiza cálculos de energia livre de perturbação (FEP) para estimar a energia livre de ligação relativa de pequenas moléculas a alvos macromoleculares, utilizando o *GROMACS*. No entanto, o *MDCrow* expande significativamente essas abordagens ao incorporar um sistema de inteligência artificial capaz de processar consultas em tempo real, recuperar informações da literatura científica e auxiliar na tomada de decisões baseadas em parâmetros específicos. A integração de LLMs em *workflows* de DM representa um avanço promissor na acessibilidade e eficiência das simulações computacionais, possibilitando que pesquisadores otimizem seus processos com maior autonomia e menor intervenção manual.

3.2 ADMIRAL

Baumgartner e Zhang (2020) [10] elaboraram o *Automated Dynamics, Molecular Interaction, and Reporting Analysis*, *ADMIRAL*, que representa um avanço significativo na automatização dos fluxos de trabalho em dinâmica molecular para a descoberta de fármacos. A ferramenta integra etapas de docking e simulações de dinâmica molecular (MD) com análises detalhadas dos resultados, automatizando não apenas a execução das simulações, mas também a compilação e interpretação dos dados. Entre os principais diferenciais do Admiral, destaca-se a capacidade de gerar automaticamente relatórios em formato de slides PowerPoint que reúnem informações cruciais, como propriedades físico-químicas, dados de docking, análises de interação proteína-ligante e imagens animadas com as trajetórias simuladas.

Essa abordagem reduz o tempo e o esforço manual comumente necessários para configurar, executar e analisar simulações DM. Ao integrar esses processos em um único fluxo contínuo, a plataforma aumenta a reprodutibilidade dos resultados. Dessa forma, o

ADMIRAL se configura como uma ferramenta capaz de ser adaptada a diferentes projetos de descoberta, contribuindo para otimização e pipelines de pesquisa em descoberta de fármacos.

3.3 GMX-QK

O artigo de Singh et al. (2023) [84] apresenta o *GMX-QK*, uma ferramenta automatizada para simulações de complexos proteína-proteína-ligante utilizando o *GROMACS*. O principal objetivo da ferramenta é simplificar a preparação e execução dessas simulações, minimizando a necessidade de intervenção manual e reduzindo erros associados à configuração dos sistemas. O *GMX-QK* integra diferentes módulos de preparação e análise, permitindo a automação do fluxo de trabalho desde a geração dos arquivos de entrada até a análise dos resultados.

A metodologia empregada pelo *GMX-QK* é baseada em um pipeline estruturado, que inclui etapas como a montagem do sistema, solvatação, adição de íons, minimização de energia, equilibração e simulação de produção. Além disso, a ferramenta incorpora funcionalidades de análise automatizada, como cálculos de energia de interação entre as biomoléculas, permitindo aos pesquisadores obter insights sobre a estabilidade e dinâmica dos complexos estudados sem a necessidade de executar comandos manualmente no *GROMACS*.

Os autores demonstram a eficácia do *GMX-QK* por meio de estudos de caso envolvendo diferentes complexos proteína-proteína e proteína-ligante, evidenciando sua aplicabilidade em diversas áreas da biologia estrutural e do desenvolvimento de fármacos. Os resultados mostram que a ferramenta reduz significativamente o tempo necessário para configurar e executar as simulações, tornando o processo mais acessível para pesquisadores que não possuem experiência avançada com o *GROMACS*.

3.4 Packmol

O artigo de Martínez et al. (2009) [63], *Packmol*, apresenta uma solução para uma das principais etapas de simulações de dinâmica molecular: a criação de configurações iniciais adequadas. O trabalho aborda o problema de *"packing"* (empacotamento) de moléculas em regiões definidas, garantindo que as distâncias mínimas entre átomos de moléculas distintas sejam respeitadas para evitar interações repulsivas abruptas, além de sobreposições atômicas que possam comprometer a estabilidade da simulação.

Uma das contribuições do *Packmol* é a capacidade de acomodar milhões de átomos agrupados em moléculas de estrutura arbitrária em domínios geométricos complexos – como interseções de esferas, elipses, cilindros, planos e caixas – com um tempo de processamento que varia de segundos a minutos em computadores pessoais.

O *Packmol* também demonstra versatilidade na construção de configurações iniciais para sistemas variados, desde a solvatação de proteínas até a montagem de estruturas ordenadas, como bicamadas lipídicas e micelas.

Ao reduzir o tempo e o esforço necessários para preparar os sistemas iniciais, o *Packmol* pode aumentar a reprodutibilidade das simulações, e facilitar a integração de configurações complexas em *workflows* automatizados.

3.5 CHAPERONg

Desenvolvido por Yekeen et al. (2023) [103], o *CHAPERONg* apresenta uma abordagem para automatizar o fluxo de simulações de DM no *GROMACS*, com foco em sistemas de proteína, proteína–ligante e proteína–DNA. O *CHAPERONg* foi desenvolvido em *Bash* e *Python*, e integra de forma direta módulos do *GROMACS* e ferramentas externas para facilitar tanto a configuração quanto a análise das simulações. Seu principal objetivo é reduzir a complexidade e a intervenção manual exigida pelos usuários, permitindo a execução de simulações completas com poucos comandos.

A ferramenta organiza o processo de forma modular, englobando a preparação do sistema até a análise pós-simulação. Durante a preparação, o *CHAPERONg* converte e valida arquivos de entrada, gera topologias de proteínas e ligantes, define a caixa de simulação, adiciona íons e realiza etapas de minimização de energia e equilíbrios *NVT/NPT*. Após a simulação, são conduzidas análises automatizadas de qualidade e de trajetória, incluindo cálculos de desvio quadrático médio (*Root Mean Square Deviation*, RMSD), flutuação quadrática média (*Root Mean Square Fluctuation*, RMSF), raio de giro, área de superfície acessível ao solvente (*Solvent-Accessible Surface Area*, SASA) e ligações de hidrogênio.

O *CHAPERONg* automatiza métodos de amostragem aprimorados, como a DM guiada e o *Umbrella Sampling*, frequentemente utilizados para estimar energias de ligação. O software também permite o cálculo da energia livre de ligação por meio do método *Molecular Mechanics Poisson–Boltzmann Surface Area* (MM-PBSA), integrando o módulo *g_mmpbsa* ao seu fluxo de trabalho. Outras ferramentas externas, como o *DSSP* (para análise de estruturas secundárias), o *PyMOL* (para visualização molecular) e o *MD DaVis* (para geração de matrizes interativas de ligações de hidrogênio e visualizações tridimensionais), são executadas automaticamente conforme a necessidade do usuário.

O *CHAPERONg* simplifica e padroniza tarefas tradicionalmente demoradas e suscetíveis a erros humanos, tornando o uso do *GROMACS* mais acessível a pesquisadores com pouca familiaridade com a linha de comando. Sua estrutura automatizada aumenta a eficiência e a reprodutibilidade das simulações, além de fornecer resultados organizados e visualmente interpretáveis, o que beneficia tanto iniciantes quanto usuários experientes em DM.

3.6 EasyAmber

O *EasyAmber* [89] é uma ferramenta desenvolvida por Suplatov, Sharapova e Švedas com o objetivo de automatizar integralmente os protocolos de simulação de DM de proteínas utilizando o pacote *Amber*. Publicada em agosto de 2020, a ferramenta foi projetada para reduzir a complexidade dos fluxos de trabalho em DM, especialmente para pesquisadores que possuem conhecimento básico em modelagem molecular, mas pouca experiência com scripts e no uso de interfaces de linha de comando.

A proposta do *EasyAmber* é simplificar as etapas do processo de simulação, desde a construção do modelo atômico completo até a execução das simulações de produção, passando pela otimização e equilíbrios do sistema. A ferramenta automatiza a preparação e a parametrização de sistemas proteína–ligante, gerando arquivos de parâmetros para ligantes a partir de modelos tridimensionais e de métodos de cálculo quântico. Essa funcionalidade reduz a necessidade de ajustes manuais e favorece a consistência entre simulações.

Além disso, o *EasyAmber* é capaz de executar protocolos avançados de simulação, incluindo dinâmicas convencionais com suporte à execução paralela de múltiplas simulações. Essa característica aumenta a eficiência computacional e o aproveitamento de recursos de hardware. Outro aspecto relevante é a redução da intervenção manual, que minimiza erros na configuração de simulações complexas.

Dessa forma, o *EasyAmber* promove uma abordagem mais dinâmica e acessível para o estudo da flexibilidade e do comportamento temporal de proteínas, contribuindo para avanços em engenharia molecular e desenvolvimento de fármacos. Ao integrar processos que antes exigiam múltiplos programas e conhecimentos avançados de programação, a ferramenta aumenta a reprodutibilidade e a escalabilidade das simulações. Nesse contexto, o *EasyAmber* se insere entre as soluções de automação que aliam robustez, reprodutibilidade e facilidade de uso, facilitando o acesso a metodologias modernas de simulação em bioinformática estrutural.

3.7 SAMSON Connect e *GROMACS* Wizard

Outra abordagem para facilitar a execução de simulações de dinâmica molecular é a plataforma integrativa SAMSON Connect [1]. SAMSON funciona como um ambiente de modelagem e simulação molecular que pode ser modificado através de extensões, distribuídas através do website SAMSON Connect [1].

A extensão ***GROMACS* Wizard** [1, 2] tem como objetivo principal integrar o pacote *GROMACS* ao ambiente gráfico do SAMSON, simplificando o processo de configuração e execução de simulações [1, 2]. O *GROMACS* Wizard guia o usuário através das etapas usuais de uma simulação de DM: preparação do sistema, minimização de energia, equilibração NVT e NPT, e a simulação de produção [1, 2].

A extensão permite o uso dos campos de força padrão distribuídos com o *GROMACS*, mas também suporta a adição de arquivos de topologia e parâmetros customizados, possibilitando a simulação de sistemas como complexos proteína-ligante ou outras moléculas parametrizadas pelo usuário [1]. Além disso, o *GROMACS* Wizard pode vir empacotado com uma versão do *GROMACS* (dispensando a instalação manual pelo usuário) ou ser configurado para utilizar uma instalação local pré-existente do *GROMACS* na máquina do usuário [2]. As simulações podem ser executadas tanto localmente quanto em ambiente de nuvem, gerenciadas pela própria plataforma SAMSON (com custos), que também facilita a importação das trajetórias e resultados gerados [1, 2].

Similarmente à proposta do LipidEase, o *GROMACS* Wizard busca reduzir a complexidade associada à linha de comando do *GROMACS* por meio de uma interface gráfica passo a passo [1, 2]. Contudo, diferencia-se por ser uma extensão dependente da plataforma SAMSON e ter um escopo mais generalista. Em contraste, o LipidEase foca especificamente na automação robusta e reprodutível da construção e simulação de **bicamadas lipídicas**, implementado como uma ferramenta independente em *Python* e com funcionalidades direcionadas à gestão detalhada de topologias lipídicas e à portabilidade para ambientes HPC (conforme descrito na Seção 4.6.6).

3.8 Comparação entre as Ferramentas Apresentadas

Para finalizar esta seção, são apresentadas duas tabelas comparativas: Tabelas 3 e 4, que sintetizam algumas das principais características das ferramentas discutidas e da proposta desenvolvida nesta pesquisa. Essas tabelas facilitam a visualização das diferenças e semelhanças entre os trabalhos.

Tabela 3: Comparação entre os programas e a ferramenta desenvolvida - Parte 1

Programa	**Linguagem**	**Interface**	**Customizável**	**Software de DM**
MDCrow	Não especificado	Linha de comando	Sim	Diversos, incl. *GROMACS*
ADMIRAL	*Python*	Linha de comando	Sim	*GROMACS*
GMX-QK	*Python*	Linha de comando	Sim	*GROMACS*
Packmol	Fortran	Linha de comando	Sim	Compatível: *GROMACS*, AMBER, LAMMPS
CHAPERONg	Não especificado	Linha de comando	Sim	*GROMACS*
EasyAmber	Não especificado	Gráfica + CLI	Sim	AMBER
SAMSON Connect	C++	Gráfica	Sim	AMBER, *GROMACS*
LipidEase	*Python*	Gráfica + CLI	Sim	*GROMACS*

Tabela 4: Comparação entre os programas e a ferramenta desenvolvida - Parte 2

Programa	**Tipo de Simulação**	**Proveniência dos Dados - Salva ou não**	**Análises dos Resultados**	**Ferramentas Externas**
MDCrow	Sistemas proteícos	Sim; armazena e recupera parâmetros e info da literatura	Indiretamente	Sim; integra +40 ferramentas
ADMIRAL	Docking, DM, Proteína-proteína-ligante	Sim; relatórios automáticos com dados e imagens	Sim; animações das trajetórias	Sim; integra docking e análise
GMX-QK	Complexos proteína-proteína-ligante	Sim; automatiza entrada até análise	Sim; análise de energia	Sim; módulos para preparação e análise
Packmol	Geração de configurações iniciais	Sim; gera arquivos iniciais de configuração	Não diretamente	Compatível com vários simuladores
CHAPERONg	Proteínas, ligantes, DNA; amostragem aprimorada	Sim; análises: RMSD, energia livre	Sim; gráficos de energia e análises	Sim; integra DSSP, PyMOL, g-mmpbsa
EasyAmber	Proteínas e complexos proteína-ligante	Sim; parametriza e gera arquivos, incl. métodos quânticos	Análises do AMBER	Sim; ferramentas de parametrização e execução paralela
SAMSON Conect (GROMACS Wizard)	Proteínas e complexos proteína-ligante	Sim; parametriza e gera arquivos	Análises de RMSD, RMSF, raio de giro	Sim, pode adicionar ferramentas externas (GROMACS Wizard)
LipidEase	Simulações de bicamadas lipídicas	Sim; automatiza geração de topologia, arquivos de simulação, parâmetros e comandos utilizados para construção do sistema	Não diretamente; depende do *GROMACS* e análises externas	Sim; requer CGenFF para parametrização inicial (Ou outro gerador de campo de força)

3.8.1 Comparação entre o LipidEase e o CHARMM-GUI

O CHARMM-GUI Membrane Builder é atualmente uma das principais plataformas para a geração automática de membranas lipídicas e preparação de sistemas para diferentes pacotes de DM, incluindo GROMACS, AMBER e NAMD [43, 102, 31]. A ferramenta permite construir bicamadas com composições complexas, inserir proteínas de membrana e gerar arquivos de entrada para diversos campos de força.

Apesar dessa versatilidade, o CHARMM-GUI apresenta limitações relevantes no contexto de automação de workflows locais:

- depende de uma interface web e de infraestrutura remota;
- não foi concebido para permitir a orquestração completa de pipelines de simulação em ambientes de linha de comando;
- não gera, de forma nativa, registros detalhados de proveniência dos parâmetros utilizados;
- não produz pacotes autocontidos voltados especificamente para execução em ambientes HPC.

O *LipidEase* complementa essas lacunas ao:

- fornecer uma solução totalmente local para preparação e execução de simulações no GROMACS;
- integrar montagem, parametrização, geração de arquivos de entrada e execução automatizada de minimização, equilíbrios e produção;
- registrar automaticamente todos os parâmetros de entrada e metadados em arquivos JSON, favorecendo rastreabilidade e reprodutibilidade [4, 3, 11];
- gerar pacotes prontos para envio e execução em clusters de alto desempenho.

Assim, enquanto o CHARMM-GUI se destaca na construção estrutural de sistemas, o *LipidEase* expande esse escopo ao oferecer um pipeline completo de automação e proveniência voltado especificamente para o ecossistema GROMACS.

A análise dos trabalhos correlatos evidencia um crescente interesse na automação de fluxos de trabalho em dinâmica molecular. Ferramentas como GMX-QK e CHAPERONg focam em sistemas proteicos, enquanto ADMIRAL se especializa na integração com docking para descoberta de fármacos. Embora o Packmol seja versátil para a montagem de sistemas, ele não automatiza todo o pipeline de simulação. Observa-se, portanto, uma lacuna específica no que tange a ferramentas dedicadas e de ponta a ponta para a montagem e execução de simulações de *bicamadas lipídicas* no *GROMACS*, que combinem uma

interface gráfica acessível com funcionalidades avançadas de reprodutibilidade e portabilidade para HPC. É precisamente nesta lacuna que o LipidEase se insere, oferecendo uma solução especializada para este importante tipo de sistema biológico.

4 MATERIAIS E MÉTODOS

O desenvolvimento da ferramenta LipidEase teve como objetivo automatizar e padronizar as etapas de preparação e execução de simulações de bicamadas lipídicas utilizando o software GROMACS. A necessidade surgiu da recorrente execução manual de comandos e do risco de erros operacionais durante a preparação dos arquivos de entrada (estruturas e topologias), montagem da bicamada, solvatação e execução das rotinas de minimização e equilíbrios. A opção por implementar a solução em Python permitiu integração com o GROMACS via chamadas de sistema, manipulação de arquivos de formato `.gro`/`.top`/`.mdp`/`.itp` e geração de arquivos de rastreabilidade, favorecendo a reprodutibilidade científica.

4.1 Linguagem de programação *Python*

Python é uma linguagem de programação interpretada, de alto nível e de propósito geral, que tem se consolidado como uma das principais ferramentas no desenvolvimento de aplicações científicas, especialmente nas áreas de bioinformática, biologia computacional e química computacional. Tem se tornado mais abrangente devido à legibilidade da sintaxe, à ampla documentação disponível e ao suporte para inúmeras bibliotecas científicas, como *NumPy, SciPy, Pandas, Matplotlib e Biopython* [70, 20].

Em relação à DM, *Python* possui capacidade de automatizar processos complexos e integrar diferentes ferramentas de simulação, como o GROMACS. Bibliotecas como MDAnalysis [65] e MDTraj [64] permitem a leitura, análise e manipulação de trajetórias moleculares, enquanto módulos nativos da linguagem viabilizam o controle do fluxo de simulações por meio de *scripts* interativos ou pipelines automatizados. A utilização desses métodos reduz significativamente a intervenção manual, minimiza erros operacionais e aumenta a reprodutibilidade das simulações, que são características essenciais na condução dos experimentos computacionais mais custosos.

Algumas ferramentas mais recentes para automação de dinâmicas moleculares, como *GMX-QK* [84] e *ADMIRAL* [10], foram desenvolvidas em *Python*, reforçando a linguagem para esse tipo de aplicação. Por sua vez, *Python* se mostra uma escolha estratégica

para o desenvolvimento da ferramenta proposta nesta dissertação.

4.2 Formatos de Arquivos

Durante a execução da ferramenta LipidEase, diversos tipos de arquivos são manipulados e gerados, cada um desempenhando um papel essencial nas etapas de preparação e execução das simulações de DM. A seguir, são descritos os principais formatos de arquivo utilizados pelo pacote GROMACS e pela ferramenta LipidEase.

4.2.1 Arquivos de Estrutura Atômica (.pdb e .gro)

Os arquivos com extensão `.pdb` (Protein Data Bank) e `.gro` (GROMOS structure file) contêm as coordenadas tridimensionais dos átomos que compõem o sistema molecular. O formato `.pdb` é amplamente utilizado em modelagem molecular por sua compatibilidade com diversas ferramentas e bases de dados estruturais, sendo geralmente usado como arquivo de entrada inicial. Já o formato `.gro` é nativo do GROMACS e apresenta uma estrutura mais compacta, incluindo informações como número sequencial de átomos, nomes de resíduos, posições cartesianas (x, y, z) e dimensões da caixa de simulação. Durante o uso do LipidEase, o arquivo `.pdb` é convertido automaticamente para `.gro` por meio do comando `gmx editconf`, garantindo compatibilidade com as rotinas subsequentes de montagem da bicamada.

4.2.2 Arquivos de Topologia (.top e .itp)

Os arquivos de topologia descrevem a conectividade e os parâmetros de interação entre os átomos do sistema. O arquivo principal, `topol.top`, contém instruções para o GROMACS sobre os tipos de átomos, ligações, ângulos, dihedros e constantes de força associadas a cada molécula. Ele também inclui referências a outros arquivos de parâmetros, que são incorporados por meio da diretiva `#include`. Os arquivos com extensão `.itp` (Include Topology) são arquivos auxiliares que definem a topologia individual de moléculas específicas, como lipídios ou ligantes. No LipidEase, o módulo `simulation.py` realiza a conversão automática de topologias oriundas do CHARMM General Force Field (CGenFF) para o formato `.itp`, e incorpora na topologia global do sistema simulado, preservando a consistência entre os parâmetros e o modelo de força selecionado, evitando conflitos de múltiplas topologias para um único sistema.

4.2.3 Arquivos de Parâmetros de Simulação (.mdp)

Os arquivos `.mdp` (Molecular Dynamics Parameters) contêm as instruções numéricas e metodológicas utilizadas pelo GROMACS para executar as diferentes etapas da simulação, como minimização de energia, equilíbrios NVT e NPT, e dinâmica de produção. Esses arquivos especificam variáveis como o passo de integração (`dt`), número

total de passos (`nsteps`), tipo de controle de temperatura e pressão, algoritmos de integração, restrições e condições de contorno periódicas. O LipidEase gera automaticamente os arquivos `.mdp` personalizados de acordo com os parâmetros definidos pelo usuário, como temperatura, tempo total de simulação e presença de íons, além de permitir a edição de cada arquivo `.mdp` gerado para modificar parâmetros mais específicos, que posteriormente são salvos no `.json` para padronização e reprodutibilidade.

4.2.4 Arquivos de Campo de Força (.ff)

Os diretórios com extensão `.ff` armazenam os parâmetros do campo de força utilizado na simulação, incluindo definições de tipos atômicos, tabelas de ligações e potenciais de interação. Esses diretórios contêm arquivos como `ffbonded.itp`, `ffnonbonded.itp`, `atomtypes.atp` e `residuetypes.dat`. Durante a preparação do sistema, o LipidEase copia automaticamente o diretório de campo de força localizado junto aos arquivos de entrada (`.pdb` e `.top`) para o diretório de trabalho da simulação, garantindo que o GROMACS tenha acesso a todos os parâmetros necessários durante a execução.

4.2.5 Arquivos de Entrada e Saída do Pré-processador (.tpr)

Os arquivos binários `.tpr` são gerados pelo pré-processador do GROMACS (`grompp`) e reúnem as informações contidas nos arquivos `.gro`, `.top` e `.mdp`. Esse arquivo único contém todos os dados estruturais, topológicos e paramétricos do sistema e é utilizado como entrada para o comando `gmx mdrun`, responsável pela execução efetiva da simulação. No fluxo do LipidEase, o arquivo `.tpr` é criado automaticamente antes de cada etapa de simulação, assegurando que alterações nos parâmetros ou na topologia sejam incorporadas de forma consistente.

4.2.6 Arquivos de Energia e Trajetória (.edr, .trr, .xtc)

Os arquivos `.edr` (energy file) armazenam informações energéticas coletadas durante a simulação, como energia potencial, energia cinética, pressão e temperatura. Já os arquivos de trajetória `.trr` e `.xtc` registram as variações das coordenadas dos átomos ao longo do tempo de simulação. Enquanto o formato `.trr` contém informações completas de precisão dupla (coordenadas, velocidades e forças), o formato `.xtc` é uma versão compactada, geralmente utilizada para análises e visualizações no VMD ou PyMOL. Esses arquivos são essenciais para análises pós-simulação, como cálculo de difusão lateral, densidade, espessura da bicamada e flutuação térmica.

4.2.7 Arquivos de Restrição de Posição

Durante as etapas de equilíbrio (NVT e NPT), é comum aplicar restrições de posição sobre átomos específicos, como os grupos de cabeça dos lipídios. Essas restrições são

descritas em arquivos `.itp`, que definem a força de restrição aplicada a cada átomo. O LipidEase permite a seleção do átomo central do lipídio para a geração dessas restrições, integrando-as automaticamente à topologia do sistema durante a preparação da simulação.

4.2.8 Arquivos de Proveniência e Relatórios (.json e .log)

Para fins de rastreabilidade e reprodutibilidade, o LipidEase gera arquivos de proveniência no formato `.json`, contendo todos os parâmetros utilizados, caminhos de arquivos, comandos executados e tempo de execução de cada etapa. Além disso, logs de saída de comandos (`.log`) e mensagens do terminal são armazenados automaticamente, permitindo a auditoria e a recuperação detalhada do histórico experimental.

4.2.9 Arquivos de Compactação e Backup

Ao término da simulação, a ferramenta também permite a compactação automática do diretório de trabalho em arquivos `.tar.gz`, contendo todos os arquivos relevantes (estruturas, parâmetros, topologias e resultados), facilitando o arquivamento e o transporte dos dados entre sistemas locais e servidores de alto desempenho.

4.2.10 Resumo dos Formatos Utilizados

A Tabela 5 resume os principais tipos de arquivos utilizados ou gerados durante a execução da ferramenta LipidEase e do GROMACS, destacando sua função no fluxo de trabalho da simulação.

Tabela 5: Principais formatos de arquivo utilizados no GROMACS e na ferramenta LipidEase.

Extensão	**Tipo de Arquivo**	**Função Principal**
`.pdb`	Estrutura atômica	Arquivo de entrada inicial
`.gro`	Estrutura GROMACS	Coordenadas e caixa do sistema
`.top`	Topologia principal	Conectividade e parâmetros moleculares
`.itp`	Topologia auxiliar	Parâmetros de moléculas individuais
`.mdp`	Parâmetros de simulação	Configuração de etapas (minim, NVT, NPT, MD)
`.ff`	Campo de força	Parâmetros físicos e químicos
`.tpr`	Pré-processado binário	Entrada para execução (`gmx mdrun`)
`.edr`	Energia	Dados termodinâmicos da simulação
`.trr`, `.xtc`	Trajetória	Registro da evolução temporal
`.posre.itp`	Restrições de posição	Controle de movimento durante equilíbrios
`.json`	Proveniência	Registro dos parâmetros e comandos executados
`.log`	Log de execução	Histórico textual das operações
`.tar.gz`	Arquivo compactado	Backup e transporte do sistema completo

4.3 Levantamento de requisitos

O levantamento de requisitos foi conduzido a partir do estudo do fluxo de trabalho típico em simulações lipídicas, tutoriais de execução, análises de trabalhos e ferramentas correlatas. A partir desse levantamento foram definidos requisitos funcionais (RF), apresentados na Tabela 6 e não funcionais (RNF), apresentados na Tabela 7, que guiaram o projeto da aplicação.

Tabela 6: Requisitos Funcionais da ferramenta LipidEase.

Código	Requisito	Descrição
RF01	Interface gráfica	Fornecer interface gráfica para a configuração dos parâmetros da simulação, evitando o uso direto do terminal.
RF02	Leitura de entradas	Permitir o carregamento, verificação e cópia automática dos arquivos de entrada do seu diretório de origem, como PDB e topologias, para o diretório de trabalho.
RF03	Construção da bicamada	Gerar monocamadas, calcular a distribuição de lipídios e combinar as estruturas para formar a bicamada lipídica de forma automatizada.
RF04	Solvatação e Neutralização	Executar a inserção de moléculas de água e íons, garantindo a integridade e neutralidade do sistema.
RF05	Adição de fármacos	Executar a inserção de moléculas de fármacos no sistema, caso necessário.
RF06	Atualização da topologia	Atualizar automaticamente o arquivo `topol.top`, inserindo corretamente as quantidades de moléculas presentes no sistema.
RF07	Execução da dinâmica	Automatizar a geração de arquivos `.mdp` e a execução sequencial das etapas de minimização, NVT, NPT e produção.
RF08	Pacote para cluster	Gerar um pacote contendo todos os arquivos necessários para execução em ambientes HPC, como clusters ou servidores remotos.
RF09	Registro de proveniência	Criar um arquivo de registro (`.json`) com parâmetros, comandos executados e informações para reprodutibilidade científica.
RF10	Tratamento de trajetória	Realizar o tratamento das trajetórias finais da fase de produção, removendo erros de periodicidade.
RF11	Análise de resultados	Realizar as análises mais comumente utilizadas em bicamadas lipídicas.

Tabela 7: Requisitos Não Funcionais da ferramenta LipidEase.

Código	Requisito	Descrição
RNF01	Usabilidade	Interface intuitiva, com validações de entrada e mensagens claras de erro/sucesso. `gui.py`.
RNF02	Portabilidade	Codificado em *Python* para execução em sistemas Linux (teste principal) e compatibilidade com outros sistemas que o GROMACS esteja instalado.
RNF03	Reprodutibilidade	Registro completo das ações e parâmetros, armazenamento de JSON de proveniência; permite reexecução.
RNF04	Modularidade	Arquitetura modular para facilitar manutenção e extensibilidade.
RNF05	Robustez	Verificações de existência de arquivos, tratamento de exceções e limpeza de arquivos temporários.
RNF06	Desempenho	Minimizar o tempo de preparação e fornecer métricas de execução para análise de performance.

4.4 Arquitetura da Ferramenta

A arquitetura adotada para o LipidEase é modular e foi projetada para separar as responsabilidades, facilitando a manutenção e a extensibilidade (RNF04). A Figura 5 ilustra esquematicamente a organização dos principais componentes do sistema e suas interações.

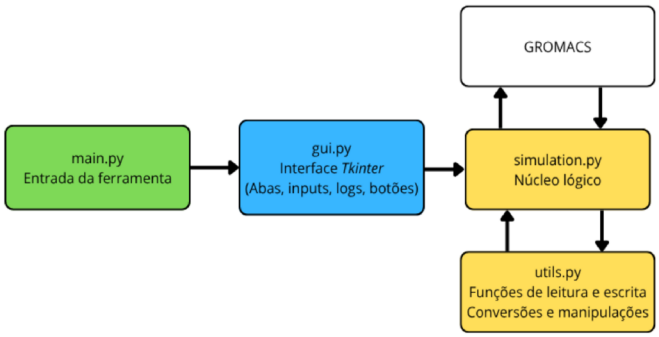


Figura 5: Diagrama de componentes da arquitetura do LipidEase, detalhando os principais módulos e suas interações no fluxo de trabalho da ferramenta. Fonte: Elaborado pelo autor (2025)

Os componentes principais são:

- `main.py`: Ponto de entrada que inicializa a aplicação e a interface gráfica.
- `gui.py`: Contém toda a camada de apresentação (Tkinter), gerenciando a

interação com o usuário, a navegação por abas, a exibição de logs e a ativação dos comandos principais.

- `simulation.py`: Implementa o núcleo da lógica de controle e orquestração. Este módulo é responsável por preparar os arquivos, gerenciar o fluxo de trabalho da simulação, executar chamadas de sistema para o GROMACS (`editconf`, `genconf`, `solvate`, `grompp`, `mdrun`), gerar o pacote para cluster e salvar o arquivo de proveniência. Ele rastreia todos os comandos e métricas de performance (RNF06).

- `utils.py`: Agrupa funções auxiliares e utilitárias para tarefas específicas, como a leitura e extração de informações de arquivos `.gro`, o cálculo da grade de lipídios, a geração de arquivos `.mdp` e a conversão de topologias `.top` para o formato `.itp`. Isso isola a lógica de manipulação de arquivos, tornando o código mais limpo e testável.

4.5 Processo de desenvolvimento

O desenvolvimento da ferramenta seguiu um modelo *iterativo e incremental*, no qual o sistema foi construído e aprimorado por meio de ciclos curtos de implementação, teste e validação [86]. Essa abordagem foi escolhida por permitir a avaliação contínua da funcionalidade da aplicação e o ajuste progressivo dos módulos conforme novas necessidades ou melhorias eram identificadas.

Cada ciclo de desenvolvimento compreendia as seguintes etapas principais:

- **Prototipagem rápida:** Foram criados scripts isolados para automatizar pequenas partes do fluxo de simulação, como a replicação de lipídios e os comandos básicos do *GROMACS* (`genconf`, `editconf`). Esses protótipos permitiram validar a viabilidade da automação antes da integração completa.

- **Modularização:** Após a validação inicial, as funcionalidades foram estruturadas em módulos independentes. As funções de manipulação de arquivos e conversão foram movidas para `utils.py`, enquanto o controle do fluxo de simulação foi centralizado em `simulation.py`. Essa separação facilitou a manutenção e o reuso de código.

- **Desenvolvimento da interface:** Foi implementada uma interface gráfica em *Tkinter* (`gui.py`), responsável por encapsular os parâmetros da simulação, reduzir erros do usuário e gerenciar a comunicação entre o usuário e o núcleo lógico da aplicação. Foram incluídas rotinas de validação de entradas e cópia automática de arquivos para o diretório de trabalho.

- **Validação funcional:** Cada incremento de código foi testado com sistemas de referência — por exemplo, a bicamada lipídica PIM_2 — comparando os resultados automatizados com execuções manuais do *GROMACS*. Essa etapa garantiu a consistência dos arquivos gerados (`.gro`, `topol.top`) e das dimensões das caixas simuladas.

- **Registro e métricas:** Ao longo do processo, foram incorporados mecanismos de registro automático, como o arquivo de proveniência (`lipidease_provenance.json`) e o monitoramento de tempos de execução dos comandos do *GROMACS*, permitindo avaliar a eficiência e a confiabilidade do sistema.

As simulações completas, incluindo as etapas de minimização, equilíbrio (NVT e NPT) e produção, foram utilizadas para verificar a robustez da aplicação e validar a execução contínua a partir de *checkpoints*. Além disso, foram realizados testes de empacotamento para execução em ambientes de alto desempenho (HPC), assegurando que todos os arquivos necessários fossem corretamente incluídos no pacote gerado para o cluster.

Essa metodologia iterativa garantiu que a ferramenta evoluísse de forma controlada, com validações progressivas e ajustes constantes, resultando em uma aplicação estável, reprodutível e cientificamente confiável.

4.6 LipidEase: Funcionalidades da ferramenta

A Figura 6 sintetiza visualmente todo o fluxo computacional do LipidEase, organizando as etapas internas da ferramenta em três grandes fases, representadas por cores distintas para facilitar a interpretação. O bloco amarelo representa a Inicialização e Preparação, na qual a ferramenta localiza o executável do GROMACS, identifica diretórios do campo de força, converte arquivos provenientes do CGenFF e aplica restrições de posição. Em seguida, o bloco azul e vermelho agrupa a fase de Construção do Sistema, onde a ferramenta gera a topologia inicial, constrói a bicamada lipídica seca, adiciona solvente, insere moléculas de fármacos quando necessário e consolida toda a estrutura final, garantindo a atualização correta das quantidades de moléculas na topologia global. Por fim, os blocos roxo e verde representam a fase de Preparação e Execução da Dinâmica, incluindo a geração automática dos arquivos `.mdp`, a neutralização opcional do sistema, a escolha do ambiente de execução (local ou HPC) e a criação dos pacotes para submissão em cluster, quando selecionado.

Além das fases, a figura explicita a função de cada módulo interno da ferramenta — derivados dos métodos da classe `SimulationManager` e da biblioteca `utils.py` — e mostra como esses módulos interagem entre si por meio de entradas, saídas e

transformações intermediárias. Cada forma geométrica possui uma função específica: os blocos arredondados indicam o início e o fim do pipeline; os retângulos coloridos representam as fases principais; os blocos cinza-azulados mostram operações atômicas do GROMACS (como `gmx editconf`, `gmx genconf`, `gmx solvate` e `gmx insert-molecules`); os diamantes rosa destacam pontos de decisão (adição de fármaco e escolha do ambiente de execução); e o cilindro lateral representa o arquivo de proveniência gerado a cada execução.

Essa organização visual evidencia que a ferramenta não é apenas um conjunto de chamadas ao GROMACS, mas um pipeline estruturado e modular, no qual cada etapa depende da integridade da anterior e alimenta automaticamente as seguintes. O uso de cores tem como objetivo orientar o leitor para o agrupamento lógico das tarefas, destacando o caráter iterativo, robusto e reprodutível do fluxo. Em conjunto, o fluxograma oferece uma visão completa do percurso computacional realizado pelo LipidEase: desde a leitura dos arquivos de entrada, passando pela preparação detalhada da bicamada e pela execução controlada da dinâmica molecular, até a geração do pacote final e o registro integral da proveniência.

4.6.1 Interface e Pré-processamento de Dados de Entrada

A interação do usuário com o LipidEase inicia-se através de uma interface gráfica projetada para capturar todos os parâmetros necessários para a simulação (**RF01**). A ferramenta inicia com os arquivos de estrutura e topologia da molécula de interesse, tipicamente obtidos a partir de servidores de parametrização externos, como o CGenFF, que gera parâmetros para o campo de força CHARMM36.

Diferentemente de um fluxo de trabalho manual, o LipidEase não requer que o usuário realize conversões prévias de formato. A ferramenta aceita diretamente os arquivos de saída padrão desses servidores, que são a estrutura em formato `.pdb` e a topologia em formato `.top`. Com esses arquivos, o LipidEase executa um pré-processamento automatizado que é fundamental para a execução da preparação de bicamadas (**RF02** e **RF06**):

1. A estrutura `.pdb` é convertida para o formato `.gro` do GROMACS através do comando `gmx editconf`.

2. A topologia `.top` é processada para extrair a definição do `[ moleculetype ]` (necessário para criar o arquivo `.itp` correto, pois contém informações da molécula), que é então convertida em um arquivo `.itp` modular. Este passo inclui a renomeação automática do resíduo, garantindo a consistência dos nomes em todo o sistema.

Além dos arquivos de entrada, a interface gráfica permite que o usuário configure os parâmetros que governam a arquitetura do sistema e as condições da simulação (**RNF01**). Esses parâmetros são validados em tempo real (existência de arquivos, tipos numéricos,

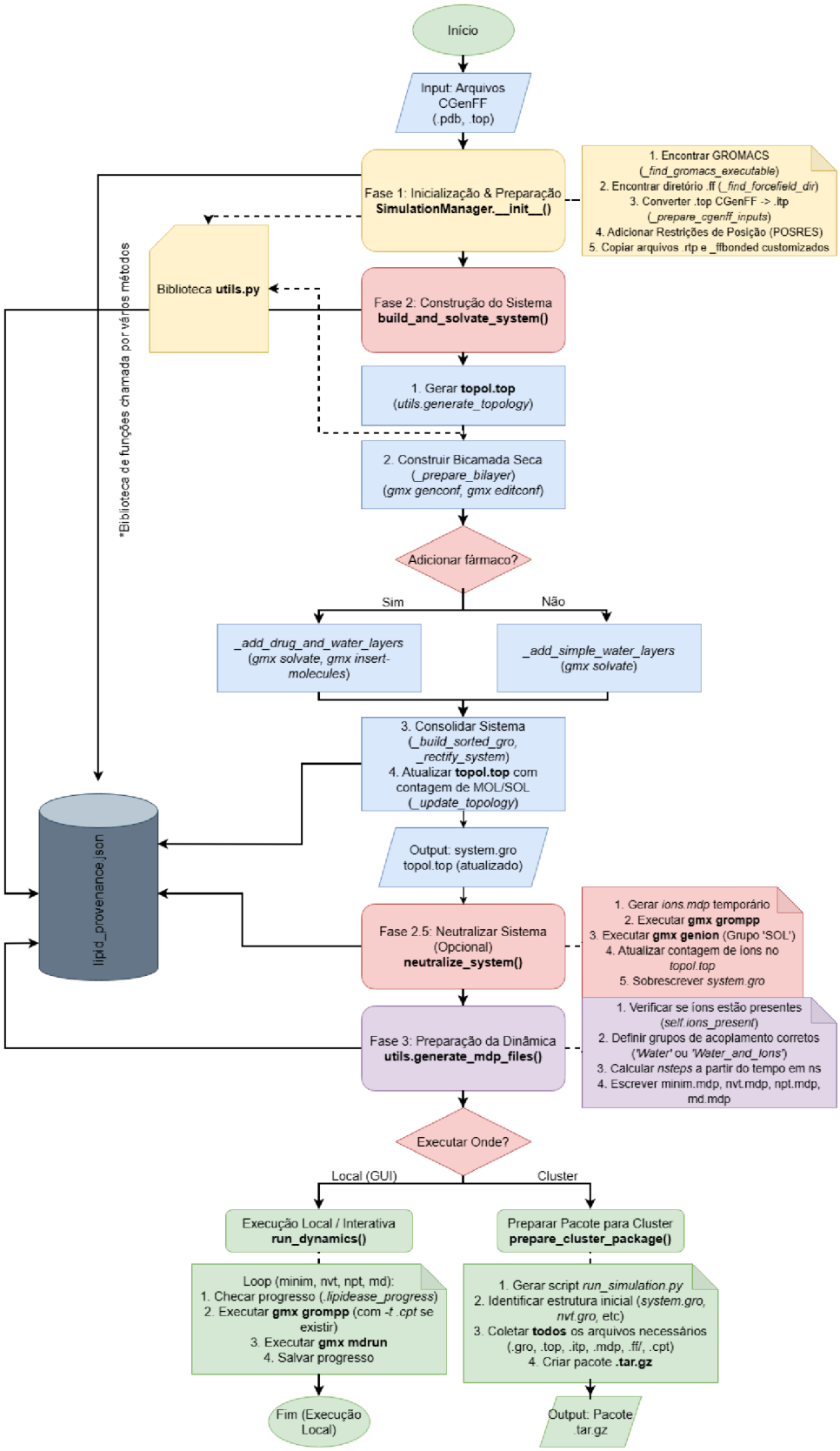


Figura 6: Fluxograma computacional da ferramenta LipidEase. As cores representam as fases principais: (1) Inicialização e Preparação (amarelo), (2) Construção do Sistema (azul/vermelho) e (3) Execução da Dinâmica (roxo/verde). Fonte: Elaborado pelo autor (2025)

faixas plausíveis) e, quando aprovados, são salvos no diretório de trabalho e registrados no arquivo de proveniência, assegurando **reprodutibilidade** e **rastreamento** (**RNF03**). A seguir, descrevem-se os principais controles expostos ao usuário:

*4.6.1.1 Número de lipídios por monocamada (lipids per layer) — **RF03**.*

Valor inteiro e positivo que determina o tamanho alvo de cada folha da bicamada. A ferramenta calcula automaticamente uma grade quase quadrada (n_x, n_y) que satisfaz $n_x \times n_y =$ `lipids_per_layer`, distribuindo os lipídios uniformemente nos eixos X e Y. Esse arranjo é usado para replicar a monocamada (via `gmx genconf`) e para estimar as dimensões iniciais da caixa nos planos X e Y (evitando sobreposição excessiva), o que melhora a estabilidade da minimização inicial.

*4.6.1.2 Tipo de caixa de simulação (box type) — **RF03/RNF05**.*

Opção do usuário (p. ex., *triclinic*, *cubic*) repassada ao GROMACS na conversão de estruturas (comando `gmx editconf`). O tipo de caixa define a geometria do volume do sistema, influenciando tanto nas condições de contorno periódicas (PBC) como no empacotamento geral do sistema.

*4.6.1.3 Espessura da camada de solvente (water box thickness, em nm) — **RF04**.*

Parâmetro numérico em nanômetros que define a altura alvo das camadas de água acima e abaixo da bicamada. A ferramenta gera duas caixas de água (superior e inferior) com essa espessura no eixo Z e concatena ao arquivo da bicamada, produzindo um sistema solvatado com caixas de solvatação simétricas.

*4.6.1.4 Átomo central para restrições de posição (position restraints) — **RF07/RNF05**.*

Seleção textual (nome do átomo no resíduo lipídico, e.g., `P` em fosfolipídios) que orienta a geração do arquivo de restrições de posição (`posre.itp`) e sua inclusão condicional na topologia. Durante NVT/NPT, as restrições estabilizam a bicamada ao limitar deslocamentos dos grupos de cabeça (ou outro átomo escolhido) enquanto o solvente relaxa na volta das moléculas, isso é necessário para evitar que o sistema se comporte erroneamente durante a introdução das condições de equilíbrio.

*4.6.1.5 Inclusão opcional de fármaco — **RF05/RF06**.*

Quando ativada, o usuário informa os arquivos da molécula (estrutura `.pdb` e topologia `.top` compatível) e a quantidade de moléculas a inserir. A ferramenta: (i) converte/normaliza a topologia para `.itp` modular, (ii) insere as moléculas nas camadas aquosas por `gmx insert-molecules`, (iii) atualiza `topol.top` com as contagens corretas e (iv) registra as alterações no JSON de proveniência.

*4.6.1.6 Validações e efeitos colaterais controlados — **RNF01/RNF05**.*

A interface realiza verificações antes de avançar para a próxima fase: existência e legibilidade dos arquivos, consistência entre nomes de resíduo e `moleculetype` na topologia, faixas válidas (valores inteiros/positivos, unidades em nm para espessura de água), além de cópia/conversão automática dos arquivos necessários para o diretório de trabalho. Em caso de inconsistência, mensagens de erro guiadas são exibidas e o estado não progride, prevenindo falhas tardias em `grompp/mdrun`.

Em conjunto, esses controles viabilizam a montagem do sistema (bicamadas com ou sem fármaco), a aplicação de restrições durante os equilíbrios e a geração de topologias consistentes, atendendo aos requisitos funcionais já definidos (**RF02**, **RF03**, **RF04**, **RF05**, **RF06**) e aos não funcionais de usabilidade, robustez e reprodutibilidade (**RNF01**, **RNF03**, **RNF05**).

4.6.2 Geração da Bicamada Lipídica

A partir dos dados fornecidos, o segundo passo de uma simulação de camada lipídica consiste em gerar a bicamada lipídica de forma automatizada (**RF03**). Esse processo ocorre em três etapas principais:

- **Preparação da monocamada** – a estrutura lipídica inicial é ajustada dentro de uma caixa de simulação, e a partir da quantidade de lipídios definidos pelo usuário, gera-se a monocamada com a replicação dos lipídios em sequência nos eixos X e Y. Caso o usuário opte por incluir uma molécula de fármaco (**RF05**), sua estrutura e topologia são carregadas nesta etapa para garantir compatibilidade com o campo de força adotado. O sistema é então preparado para a inserção posterior dessas moléculas, respeitando os limites geométricos definidos pela caixa simulada.

- **Criação da bicamada** – a segunda monocamada é gerada por inversão da estrutura original, de forma que as regiões hidrofóbicas fiquem voltadas uma para a outra, e ambas são combinadas para formar a bicamada lipídica completa. Se houver fármacos definidos, a ferramenta armazena as coordenadas livres nas regiões da caixa de água adjacentes à bicamada, onde essas moléculas serão inseridas durante a solvatação.

- **Solvatação** – as moléculas de solvente são adicionadas às regiões hidrofílicas do sistema (**RF04**), utilizando o tamanho de caixa d'água no eixo Z definido pelo usuário. As dimensões nos eixos X e Y são automaticamente calculadas pela ferramenta, assegurando a integridade estrutural do sistema. Nesta etapa, caso o sistema contenha fármacos, as moléculas são posicionadas de forma aleatória nas regiões aquosas através do comando `gmx insert-molecules`, garantindo distância mínima entre o fármaco, as moléculas de solvente e os lipídios. Após a inserção, a

topologia global é atualizada para refletir as quantidades corretas de cada componente, assegurando a consistência dos arquivos estruturais e topológicos (**RF06**).

Todo esse processo ocorre levando em conta possíveis sobreposições de átomos, utilizando as coordenadas máximas e mínimas de cada componente do sistema para realizar o ajuste sem colisões, atendendo ao requisito de robustez (**RNF05**).

4.6.3 Execução dos Passos da Dinâmica Molecular

Após a configuração do sistema, a ferramenta conduz automaticamente as etapas da dinâmica molecular no GROMACS (**RF07**), em sequência controlada:

- **Minimização de Energia** – remove colisões e relaxa a estrutura inicial do sistema.
- **Equilíbrio** – conduzida em duas fases com restrição de movimento:
  - Equilíbrio NVT: mantém temperatura e volume constantes.
  - Equilíbrio NPT: mantém temperatura e pressão constantes, permitindo variação do volume.
- **Dinâmica de Produção** – executada pelo tempo definido pelo usuário, permitindo a análise das propriedades da bicamada.

A execução de cada etapa é monitorada pela ferramenta, que verifica a conclusão correta de cada fase antes de prosseguir (**RNF05** e **RNF06**). Isso garante a integridade do processo e permite o registro completo para reprodutibilidade (**RNF03**).

4.6.4 Justificativa dos tempos de equilíbrio

Protocolos de equilíbrio em DM podem variar em função do tamanho do sistema, do campo de força empregado e do tipo de membrana estudada. No entanto, equilíbrios na ordem de 1 ns para as etapas NVT e NPT são amplamente empregados em sistemas lipídicos de tamanho moderado, especialmente quando restrições posicionais são aplicadas às cabeças polares nas etapas iniciais [30, 75, 85].

Esse procedimento permite que:

- água e íons acomodem-se adequadamente no volume da caixa;
- as caudas lipídicas relaxem sem distorções abruptas da bicamada;
- a área por lipídio e a espessura convirjam para valores estáveis;
- temperatura, pressão e volume alcancem equilíbrio em escalas de tempo compatíveis com as recomendações de boas práticas para membranas [85].

Os de Apl apresentados neste trabalho indicam que esses parâmetros se estabilizam dentro do intervalo utilizado, justificando a adoção de 1 ns para NVT e 1 ns para NPT nas simulações conduzidas.

4.6.5 Saída e Pós-Processamento

Ao término da simulação, o GROMACS gera automaticamente os arquivos de saída de energia, coordenadas e trajetórias (`.gro`, `.trr`, `.xtc`, `.edr`, `.log`), que são organizados pela ferramenta para análise e arquivamento (**RF10**, **RF11**). Essa estrutura automatizada garante a execução com mínima intervenção manual, otimizando o fluxo de trabalho e reduzindo o risco de erros humanos (**RNF01**, **RNF05**).

Embora a ferramenta já realize a organização dos resultados e o registro das informações de proveniência, as rotinas de tratamento de trajetórias e análises pós-simulação ainda não foram totalmente implementadas na versão atual do LipidEase. Tais funcionalidades, correspondentes aos requisitos funcionais **RF10** e **RF11**, estão previstas para versões futuras e incluirão o processamento automatizado das trajetórias finais (remoção de periodicidade, alinhamento e centralização do sistema) e a execução de análises estruturais e dinâmicas comuns em bicamadas lipídicas, como espessura, área por lipídio e difusão lateral.

4.6.6 Reprodutibilidade e rastreabilidade

A cada execução é gerado o arquivo `lipidease_provenance.json`, contendo: versão da ferramenta, usuário, *timestamp* UTC, diretório de trabalho, versão do GROMACS utilizada, parâmetros de simulação, comandos executados, *logs* e métricas de desempenho (**RF09** e **RNF03**).

Os *logs* registram todas as chamadas realizadas ao GROMACS, incluindo mensagens do terminal, comandos executados e eventuais erros. As métricas de desempenho armazenam o tempo total da simulação, o tempo gasto em cada etapa (minimização, equilíbrios NVT e NPT, e dinâmica de produção), além de informações sobre a versão e ambiente de execução. Esses dados são consolidados no arquivo `lipidease_provenance.json`, viabilizando a auditoria dos resultados e a repetição exata do experimento em diferentes ambientes computacionais.

4.6.6.1 Gerenciamento Automatizado de Topologias

O tratamento de arquivos de topologia, especialmente os gerados por servidores web como o CGenFF, é automatizado pela ferramenta (**RF06**). O LipidEase lê o arquivo `.top`, extrai a diretiva `[ moleculetype ]`, renomeia o resíduo e gera um arquivo `.itp` modular, além de copiar automaticamente arquivos complementares de parametrização, garantindo consistência entre topologias e campos de força (**RNF05**).

4.6.6.2 Garantia de Reprodutibilidade e Proveniência de Dados

Para atender aos princípios FAIR e aos requisitos de reprodutibilidade científica (**RNF03**), o LipidEase gera, a cada execução, um arquivo de registro completo chamado `lipidease_provenance.json`, um arquivo exemplo desses parâmetros salvos pode

ser encontrado no Anexo B na Seção 7, nas Figuras 37, 38, 39, 40 e 41. Esse registro contém todos os parâmetros, arquivos de entrada, comandos executados e métricas de execução, permitindo replicar o experimento com exatidão.

4.6.6.3 Portabilidade para Ambientes de Alta Performance (HPC)

Visando suportar a execução em ambientes de alto desempenho (**RF08** e **RNF02**), a ferramenta implementa a funcionalidade "Preparar Execução em Cluster". Esse recurso empacota todos os arquivos essenciais em um único `.tar.gz` e gera um *script Python* pronto para execução remota, eliminando a necessidade de ajustes manuais e promovendo maior portabilidade entre sistemas locais e HPCs.

4.6.7 Funcionalidades, vantagens e limitações

De forma integrada, as funcionalidades descritas nesta seção mostram que o LipidEase vai além de um conjunto de *scripts* auxiliares para o GROMACS. A ferramenta combina: (i) uma interface gráfica com validação de entradas e registro automático de parâmetros (**RF01**, **RF02**, **RNF01**); (ii) a geração e atualização consistentes de estruturas, topologias e arquivos de parâmetros, incluindo a conversão de topologias oriundas do CGenFF para o formato modular `.itp` (**RF03**, **RF04**, **RF05**, **RF06**); (iii) a execução encadeada das etapas de minimização, equilíbrios NVT/NPT e dinâmica de produção (**RF07**); (iv) o empacotamento automático para execução em ambientes de alto desempenho (**RF08**, **RNF02**); e (v) o registro detalhado de proveniência em formato `.json`, com métricas de desempenho e histórico completo de comandos (**RF09**, **RNF03**, **RNF06**). Em conjunto, esses componentes configuram um pipeline único e padronizado para simulações de bicamadas lipídicas no GROMACS, reduzindo a intervenção manual e mitigando erros operacionais recorrentes na preparação de sistemas.

Por outro lado, a versão atual do LipidEase ainda não implementa de forma completa as rotinas de tratamento automatizado de trajetórias e de análises pós-simulação previstas nos requisitos **RF10** e **RF11**. Embora a ferramenta organize os arquivos de saída e registre todas as informações necessárias para a reprodutibilidade, o cálculo de propriedades estruturais e dinâmicas da bicamada (como espessura, área por lipídio e coeficientes de difusão) ainda depende de comandos externos e do uso direto do GROMACS ou de outras bibliotecas de análise. Essas limitações definem um espaço claro para evolução futura da ferramenta, em que módulos adicionais poderão automatizar também a etapa analítica do fluxo de trabalho.

5 RESULTADOS

Este capítulo apresenta os resultados obtidos com o desenvolvimento e a aplicação da ferramenta LipidEase para diferentes estudos de caso. O objetivo é demonstrar não apenas a funcionalidade da interface gráfica e do fluxo de trabalho automatizado, mas também validar cientificamente a robustez dos sistemas de bicamadas lipídicas gerados.

A apresentação dos resultados está estruturada em três seções principais. Primeiramente, será detalhada a interface gráfica da ferramenta, destacando como sua arquitetura foi projetada para guiar o usuário de forma intuitiva através das etapas de configuração e execução da simulação. Em seguida, será apresentado um estudo de caso detalhado com o lipídio PIM_2, ilustrando visualmente cada etapa do processo automatizado de construção do sistema — desde a criação da monocamada até a solvatação e o equilíbrio.

Por fim, e de maneira crucial para a validação do trabalho, serão apresentados os resultados quantitativos da aplicação da ferramenta na construção de múltiplos sistemas lipídicos padrão na literatura (como DPPC, POPC e DOPC).

5.1 CHARMM-GUI x LipidEase

A fim de contextualizar o desempenho do LipidEase, foi realizada uma comparação direta com o CHARMM-GUI, atualmente a ferramenta mais utilizada para a construção automatizada de sistemas biomoleculares. Enquanto o CHARMM-GUI executa a geração completa de uma bicamada lipídica de forma remota, distribuída em várias etapas e com dependência de comunicação com o servidor, o LipidEase produz a estrutura final pronta para minimização de energia localmente em aproximadamente 10 segundos (variando com o tamanho do sistema, sistemas que necessitam de caixas de água maiores, levam mais tempo devido a calculos do GROMACS), incluindo a montagem da bicamada, solvatação, geração dos arquivos `.top`, `.itp` e `.mdp`, e organização dos diretórios de execução. Em comparação, o fluxo completo do CHARMM-GUI para um sistema de complexidade equivalente demanda tipicamente entre 4 a 8 minutos, dependendo da carga do servidor e da necessidade de etapas interativas de configuração. Assim, o LipidEase apresenta uma redução de uma ordem de magnitude no tempo de preparação, além de

eliminar dependências externas de infraestrutura e a necessidade de etapas manuais.

Tabela 8: Comparação entre o LipidEase e o CHARMM-GUI para a construção de bicamadas lipídicas.

Critério	**LipidEase**	**CHARMM-GUI**
Tempo de construção	**∼10 s**	**4–8 min**
Dependência de servidor externo	Não	Sim
Preparação totalmente local	Sim	Não
Etapas interativas	Mínimas	Diversas
Geração automática de `.mdp`/`.top`/`.itp`	Sim	Parcial
Empacotamento para HPC	Sim	Não
Registro automático de proveniência	Sim	Não
Foco em membranas lipídicas	Total	Parcial

5.2 Interface Gráfica

Com o uso do pacote *Tkinter* em *Python*, foi desenvolvida uma interface gráfica, projetada para guiar o pesquisador através das etapas lógicas de uma simulação de DM de bicamadas lipídicas. A ferramenta, LipidEase, organiza o fluxo de trabalho em três abas sequenciais — **1. Definição de Parâmetros**, **2. Montagem do Sistema** e **3. Dinâmica Molecular** — que se habilitam progressivamente, minimizando erros operacionais e garantindo que a sequência correta de comandos seja executada.

5.2.1 Aba 1: Definição de Parâmetros (*Parameter Definition)*

A primeira aba centraliza todos os parâmetros e dados de entrada necessários para a configuração do sistema a ser simulado (Figura 7). As funcionalidades incluem:

- **Reprodutibilidade e Carregamento:** O usuário pode iniciar uma nova simulação ou carregar todos os parâmetros de uma execução anterior selecionando um arquivo de proveniência (salvo em outra execução) `lipidease_provenance.json`, o que garante a total reprodutibilidade dos experimentos.

- **Diretório de Trabalho:** É possível definir um diretório de trabalho (*working directory*) dedicado para a simulação, que armazenará todos os arquivos gerados, assegurando a organização do projeto.

- **Arquivos de Entrada:** A interface permite a seleção dos arquivos de estrutura `lipid structure.pdb` e topologia `lipid topology.top` para o lipídio. Uma funcionalidade de conveniência foi implementada para tentar localizar automaticamente o arquivo `.top` correspondente quando um `.pdb` é selecionado.

- **Sistema Complexo (Opcional):** A ferramenta oferece suporte nativo à adição de uma segunda molécula, como um fármaco, ao sistema. Ao selecionar a opção `Add`

`drug molecule to the system`, campos adicionais para a estrutura, topologia e quantidade de moléculas de fármaco são habilitados.

- **Parâmetros da Simulação:** Menus suspensos e campos de texto permitem a fácil configuração de parâmetros cruciais, como a quantidade de lipídios por camada, o tipo de caixa de simulação (*triclinic*, *cubic*, etc.), a espessura da camada de água, a temperatura do sistema e o tempo total da simulação de produção.

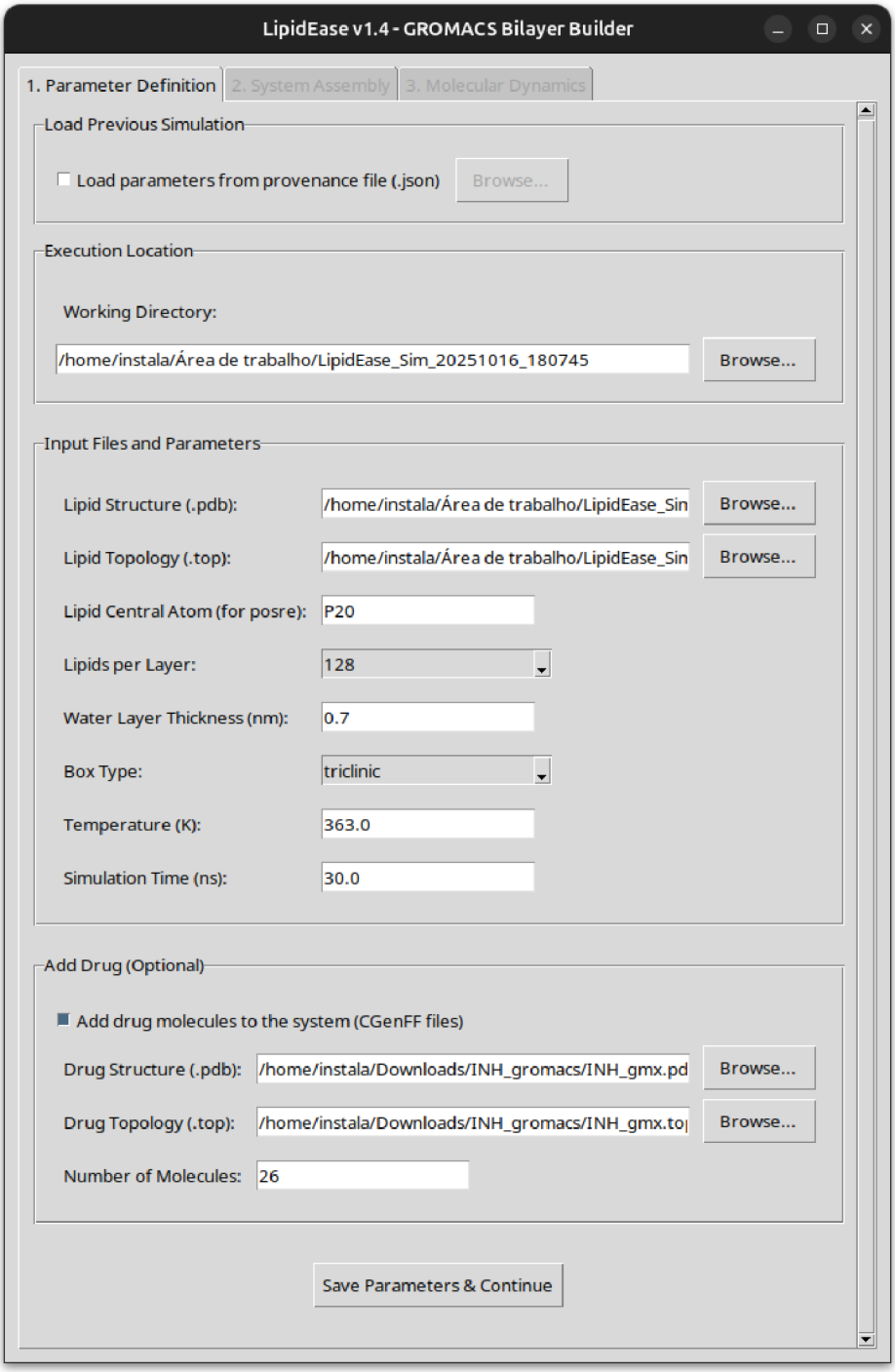


Figura 7: Interface gráfica desenvolvida em *Python* com *Tkinter* para configuração dos parâmetros de simulação (Aba 1 - Seleção de parâmetros da simulação).

5.2.2 Aba 2: Montagem do Sistema (*System Assembly)*

Uma vez que os parâmetros são salvos, a segunda aba é habilitada (Figura 8), focando na construção e preparação do sistema para simulação de DM. Ela contém um fluxo de trabalho em duas etapas:

1. **Construir e Solvatar Sistema (*Build & Solvate System*):** Este comando inicia o processo automatizado de construção da bicamada, solvatação com água e, se aplicável, inserção das moléculas de fármaco.

2. **Neutralizar com Íons (*Neutralize with Ions*):** Após a construção, este botão (inicialmente desabilitado) torna-se ativo, permitindo que o usuário neutralize a carga do sistema com a adição de íons.

Um console de saída de texto exibe em tempo real os logs dos comandos do *GROMACS*, oferecendo total transparência sobre o processo e facilitando a identificação de eventuais problemas.

5.2.3 Aba 3: Dinâmica Molecular (*Molecular Dynamics*)

A terceira aba concentra os mecanismos de execução da Dinâmica Molecular (Figura 9), representando o núcleo operacional da ferramenta. Esta etapa foi projetada para oferecer ao usuário tanto controle sobre o fluxo da simulação quanto robustez no gerenciamento de estados, integrando práticas de automação e reprodutibilidade.

- **Seleção de Etapas:** O usuário pode escolher quais fases da simulação executar (Minimização de Energia, Equilíbrio NVT, Equilíbrio NPT e Produção MD) permitindo desde execuções completas até execuções parciais. Essa flexibilidade é particularmente útil para depuração, análises intermediárias ou reinicializações controladas.

- **Geração e Edição de Arquivos de Parâmetros:** Os arquivos de parâmetros (`.mdp`) de cada etapa são gerados com um clique, com base nas configurações definidas nas abas anteriores e parâmetros comumente utilizados em DM de bicamadas lipídicas, utilizando o módulo interno `utils.py`. Cada arquivo é criado de forma modular, garantindo consistência entre diferentes execuções e compatibilidade com o GROMACS. Usuários avançados podem abrir e editar esses arquivos diretamente da interface, permitindo ajustes finos a parâmetros específicos da simulação.

- **Execução Local e Preparação para HPC:** Considerando que simulações de produção exigem alto poder computacional, o LipidEase oferece o recurso **"Preparar Execução em Cluster"**. Essa funcionalidade empacota automaticamente todos os arquivos essenciais (incluindo estruturas, topologias, parâmetros e o campo

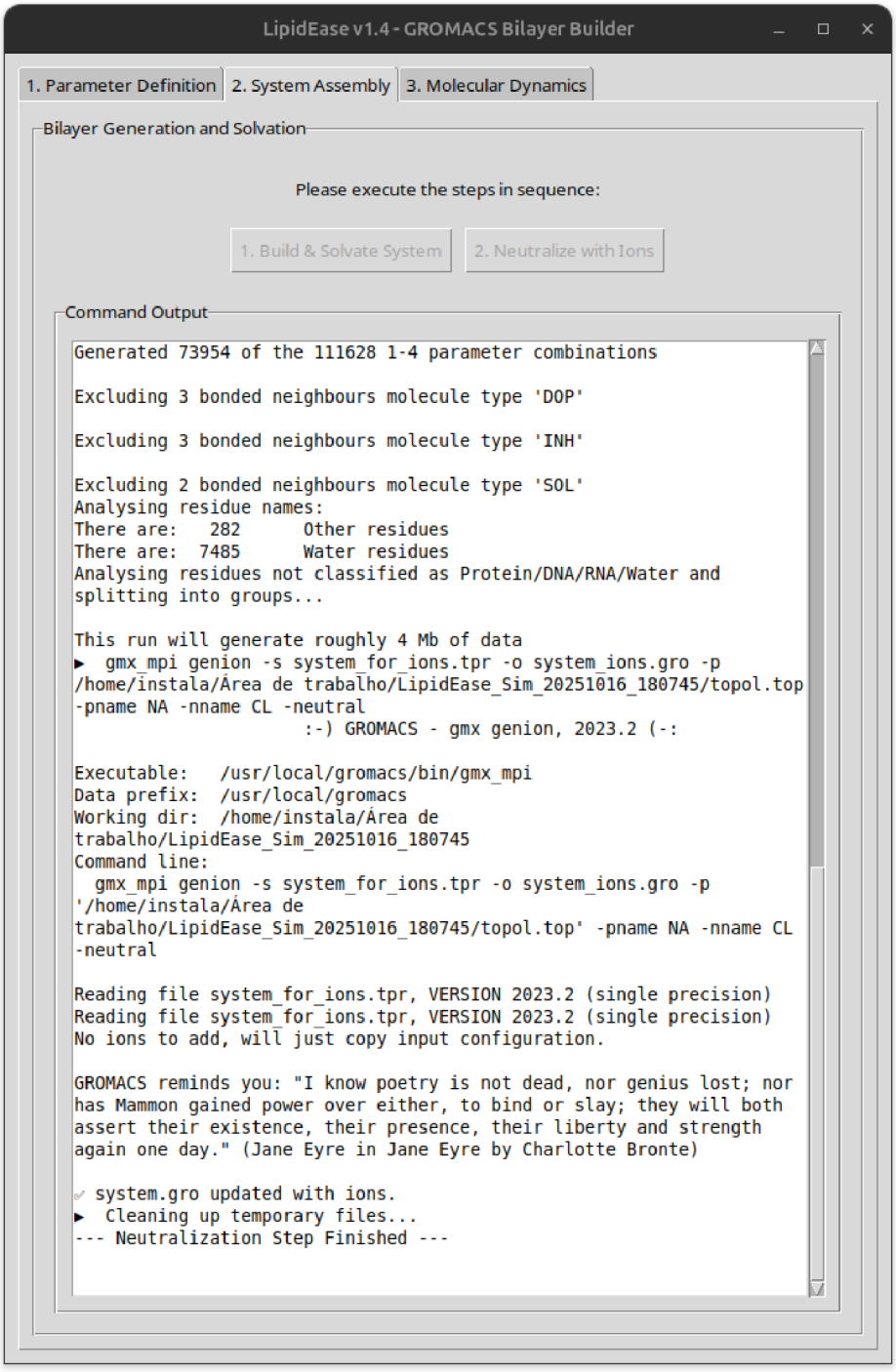


Figura 8: Interface gráfica desenvolvida em *Python* com *Tkinter* (Aba 2 - Solvatação e Neutralização do sistema).

de força) em um arquivo compactado `.tar.gz`. O pacote inclui ainda o script `run_simulation.py`, que automatiza a execução sequencial das etapas no ambiente de cluster, garantindo compatibilidade com sistemas de agendamento e filas (*job schedulers*). Esse empacotamento assegura que o experimento possa ser reproduzido em diferentes ambientes computacionais sem necessidade de reconfiguração manual.

- **Controle de Execução e Continuidade:** O LipidEase implementa um sistema de rastreamento interno para manter o estado da simulação. A cada execução, é gerado um arquivo oculto (`.lipidease_state`) e um log de execução que registram qual etapa foi concluída com sucesso. Caso o processo seja interrompido, a ferramenta utiliza esses registros — em conjunto com os arquivos de ponto de verificação (`.cpt`) do GROMACS — para identificar a última etapa completada e retomar automaticamente a execução a partir desse ponto.

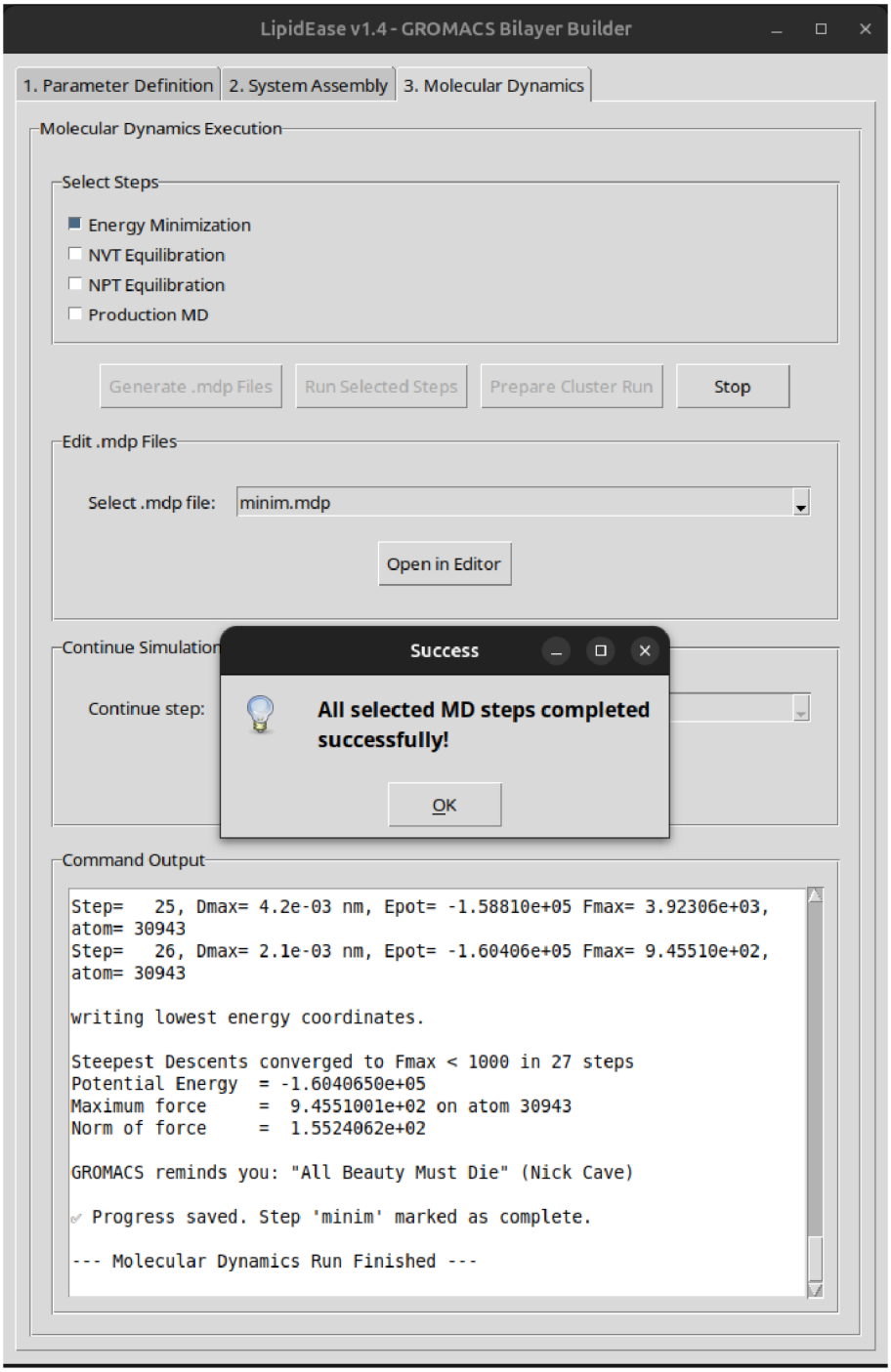


Figura 9: Interface gráfica desenvolvida em *Python* com *Tkinter* (Aba 3 - Execução dos passos de dinâmica molecular, geração de *script* para *cluster*).

5.3 Validação da Ferramenta com Múltiplos Sistemas Lipídicos

Para validar a robustez e a eficácia do protocolo de montagem e equilíbrio implementado no LipidEase, a ferramenta foi utilizada para gerar e simular um total de nove sistemas distintos, separados a seguir em dois estudos de caso. Este conjunto de testes foi projetado para cobrir uma gama diversificada de cenários, incluindo três variações do lipídio PIM_2: um sistema puro e dois sistemas contendo os fármacos benzohidroxamato (BH) e nitrobenzohidroxamato (NBH), além de seis fosfolipídios padrão (DPPC, DLPE, DMPC, DOPC, POPC e POPE). O objetivo desta análise é demonstrar que o fluxo de trabalho automatizado produz sistemas estruturalmente estáveis sob diferentes condições, um pré-requisito fundamental para qualquer simulação de produção.

Os parâmetros específicos para cada uma das nove simulações de validação, todos configurados através da interface do LipidEase, estão sumarizados na Tabela 9.

Tabela 9: Parâmetros de simulação para os seis sistemas de lipídios comuns utilizados na validação da ferramenta LipidEase.

Sistema	**Lipídio**	**Fármaco (Nº Moléculas)**	**Nº Lipídios**	**Espessura H_2O (nm)**	**Temp. (K)**	**Tempo (ns)**
Estudo de Caso 1	PIM_2	Nenhum	128	0.8	310	100
Estudo de Caso 1	PIM_2	Benzohidroxamato (12)	128	0.8	310	100
Estudo de Caso 1	PIM_2	Nitrobenzohidroxamato (12)	128	0.8	310	100
Estudo de Caso 2	DPPC	Nenhum	512	0.4	310	10
Estudo de Caso 2	DLPE	Nenhum	128	1.0	310	30
Estudo de Caso 2	DMPC	Nenhum	128	1.5	303	30
Estudo de Caso 2	DOPC	Nenhum	256	0.7	363	30
Estudo de Caso 2	POPC	Nenhum	64	1.1	340	30
Estudo de Caso 2	POPE	Nenhum	256	0.7	298	30

5.4 Estudo de caso 1: Simulações com lipídio PIM_2

O modelo lipídico PIM_2 (*fosfatidil-mio-inositol dimanosídeo*) foi desenvolvido e validado em trabalhos anteriores conduzidos por membros do mesmo grupo de pesquisa [82]. Esse lipídio é um dos principais constituintes da membrana plasmática de *Mycobacterium tuberculosis*, representando até 42% dos lipídios totais da membrana interna, e está diretamente associado à resistência intrínseca dessa bactéria a diversos fármacos. Assim, sua escolha como estudo de caso nesta dissertação justifica-se tanto pela importância biológica quanto pela viabilidade computacional já demonstrada em estudos anteriores, servindo como um sistema para validar a robustez e a aplicabilidade da ferramenta LipidEase. Nessa seção, será abordada também o passo-a-passo da construção de um sistema lipídico PIM_2, sem fármacos.

5.4.1 Caixa de simulação estrutura lipídica

Inicialmente, a ferramenta determina as coordenadas limitantes (mínimas e máximas) (Pseudocódigo localizado no Anexo A, algoritmo 6.1.8) dos átomos do lipídio, como pode

ser observado na Figura 10, a partir do arquivo de entrada, para definir as dimensões de uma caixa de simulação que o envolva precisamente. Esta etapa é crucial para a subsequente replicação da molécula na formação da monocamada, pois uma delimitação precisa da caixa previne sobreposições entre átomos durante o processo de construção do sistema.

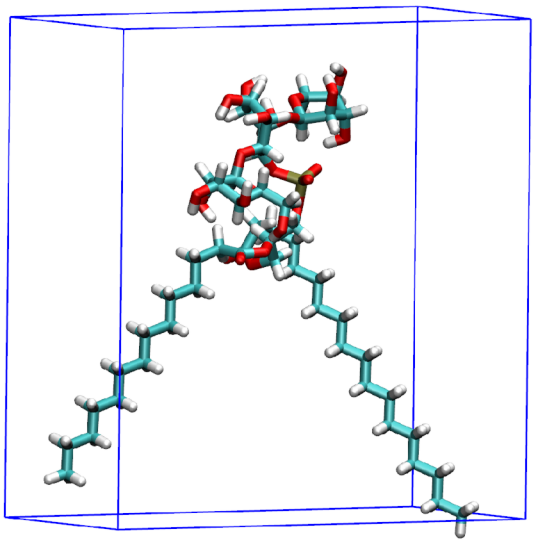


Figura 10: Primeira etapa executada pela ferramenta: Definição dos vetores da caixa de simulação para um lipídio.

5.4.2 Criando monocamadas

Nessa etapa, utilizando comandos internos do *GROMACS*, a ferramenta multiplica a quantidade de lipídios no eixo X e Y (Pseudocódigo 6.1.9), como observado na Figura 11, para atingir a quantidade definida de lipídios por monocamada pelo usuário.

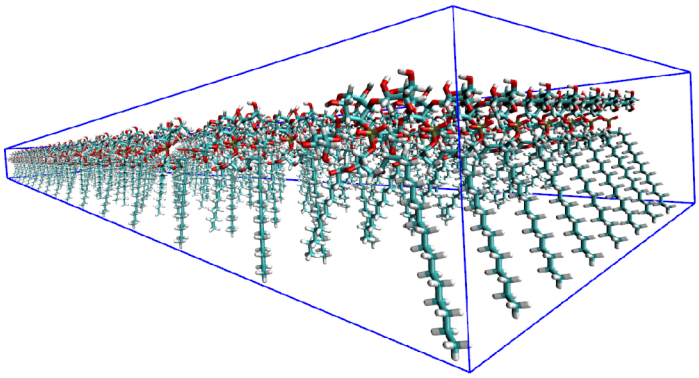


Figura 11: Criação da monocamada lipídica com 64 lipídios, replicando o primeiro lipídio nos eixos X e Y, na quantidade definida pelo usuário.

5.4.3 Concatenação das monocamadas

Posteriormente à formação da monocamada pela replicação das moléculas lipídicas, a ferramenta gera uma segunda monocamada. Esta é obtida através da inversão da primeira, em um novo arquivo.

Para construção da bicamada, a posição da monocamada invertida é calculada automaticamente (Pseudocódigos 6.1.8,6.1.10), para realizar a transladação das monocamadas anterior à concatenação dos arquivos. Esse cálculo considera as dimensões da monocamada original (particularmente seus limites ao longo do eixo perpendicular ao plano da camada, tipicamente o eixo Z) e um espaçamento predefinido, e após concatenados, temos uma bicamada pronta como na Figura 12.

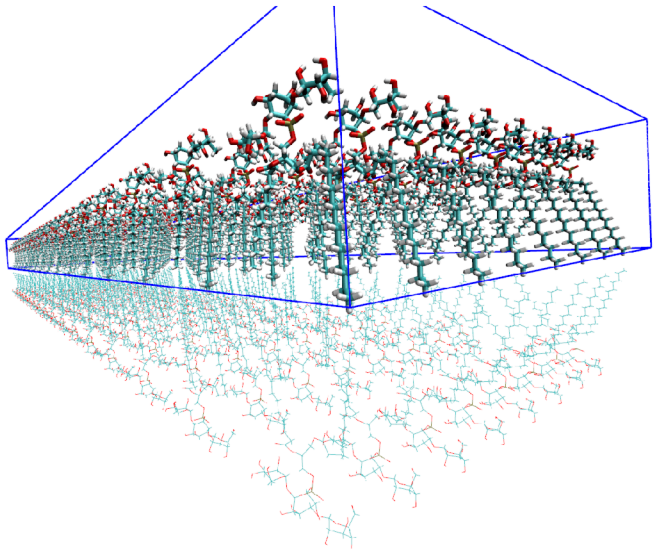


Figura 12: Concatenação dos arquivos de monocamadas para formação da bicamada lipídica, com espaçamento de 0,1 nm entre elas no eixo Z. Contém diferença de representação visual para enfatizar as diferentes estruturas.

5.4.4 Sistema de bicamada pronto

Esta subseção demonstra a configuração da bicamada lipídica resultante do processo de concatenação dos arquivos. Conforme ilustrado na Figura 13, o sistema exibe as monocamadas corretamente posicionadas e uma estrutura geral com a morfologia esperada, sem anomalias visuais. Nota-se o pequeno espaço projetado entre as superfícies das duas monocamadas, o qual é fundamental para prevenir colisões atômicas entre as camadas opostas.

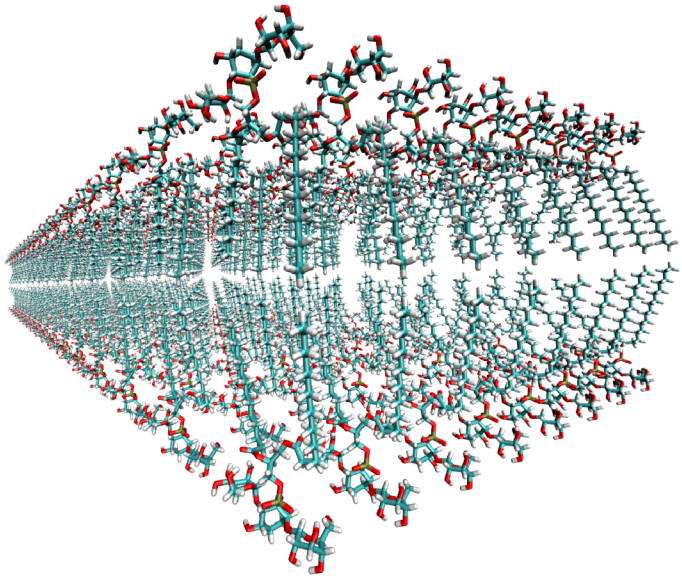


Figura 13: Bicamada com 128 moléculas de lipídios PIM_2 construída.

5.4.5 Solvatação

A etapa de solvatação do sistema é configurada a partir de duas especificações do usuário: o modelo de água a ser empregado (por exemplo, TIP3P, TIP4P, SPC/E, SPC216) e a espessura da camada de solvatação desejada em cada lado da bicamada ao longo do eixo Z (em nanômetros) .

Inicialmente, são preparadas duas caixas de água (uma superior, como na Figura 14 e outra inferior). As dimensões laterais (X e Y) dessas caixas de água são determinadas pelas coordenadas da bicamada lipídica previamente montada através da determinação dos valores mínimos e máximos dos eixos X e Y, enquanto a espessura no eixo Z corresponde ao valor fornecido pelo usuário para cada camada de solvatação.

Subsequentemente, essas duas camadas de água são translocadas para as regiões adjacentes às superfícies da bicamada (próximas às cabeças dos lipídios). As posições exatas são calculadas com base nas coordenadas Z mínima (para a caixa inferior) e máxima (para a caixa superior) da bicamada. Este procedimento assegura que, ao concatenar os arquivos de coordenadas da bicamada e das duas camadas de água, não ocorram sobreposições estéricas (colisões atômicas) entre as moléculas.

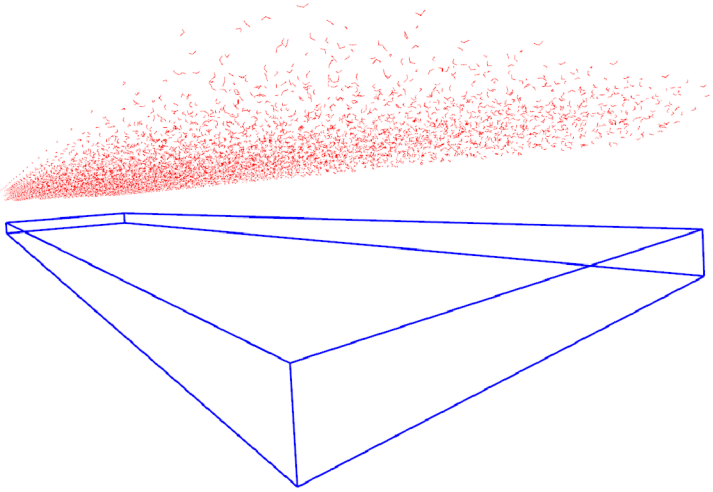


Figura 14: Caixa de água superior na espessura definida por usuário (eixo Z), do tamanho da bicamada nos eixos X e Y através da obtenção de coordenadas máximas e mínimas do sistema. A caixa de simulação está deslocada abaixo, em azul, para demonstrar a posição original da caixa de água, antes do translado.

5.4.6 Protocolo de equilíbrio e simulação

A etapa de equilíbrio do sistema é conduzida por uma sequência de comandos do *GROMACS*, iniciando-se com o processo de minimização de energia. A minimização de energia é um procedimento computacional que visa refinar a estrutura tridimensional inicial do sistema molecular. Durante este processo, os átomos são sutilmente reposicionados de forma iterativa, buscando uma configuração que corresponda a um mínimo local na superfície de energia potencial do sistema, conforme definida pelo campo de força utilizado. Essencialmente, o algoritmo de minimização (como o *steepest descent* ou o *conjugate gradient*, comumente usados) calcula as forças sobre cada átomo e os desloca em direções que reduzam a energia potencial total do sistema. É importante notar que esta é uma otimização geométrica e não envolve a simulação da dinâmica real ao longo do tempo; não há atribuição de velocidades ou controle de temperatura nesta fase, podemos observar o sistema minimizado na Figura 15.

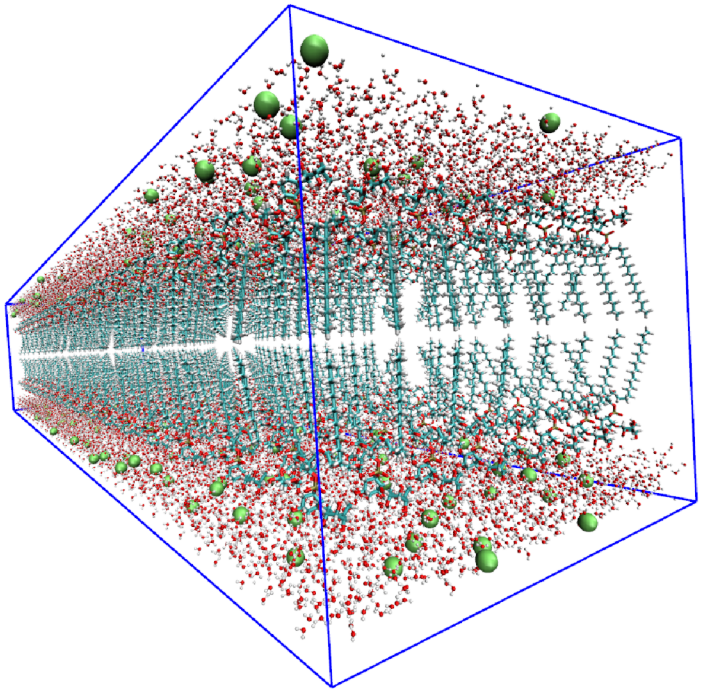


Figura 15: Sistema de bicamada lipídica construído e minimizado, com solvente e íons presentes. Os íons foram aumentados durante a visualização para confirmar sua presença no sistema.

A primeira fase do equilíbrio foi conduzida sob o ensemble NVT (Número de partículas, Volume e Temperatura constantes). O objetivo principal desta etapa é estabilizar a temperatura do sistema no valor desejado e permitir que as moléculas de solvente se acomodem ao redor da bicamada lipídica, que ainda possui suas posições restringidas. Conforme observado na Figura 16, ao final desta etapa, o sistema atinge o equilíbrio térmico. É importante notar, como indicado na legenda da Figura 16 e visível na imagem, que os lipídios foram mantidos com restrições de movimento ao longo do eixo Z. Essas restrições são recomendadas durante as fases de minimização e equilíbrio para preservar a integridade e a organização geral da bicamada enquanto o solvente se reorganiza e a temperatura do sistema se estabiliza, evitando deformações excessivas ou o colapso da estrutura da membrana antes que o sistema atinja uma densidade apropriada [85].

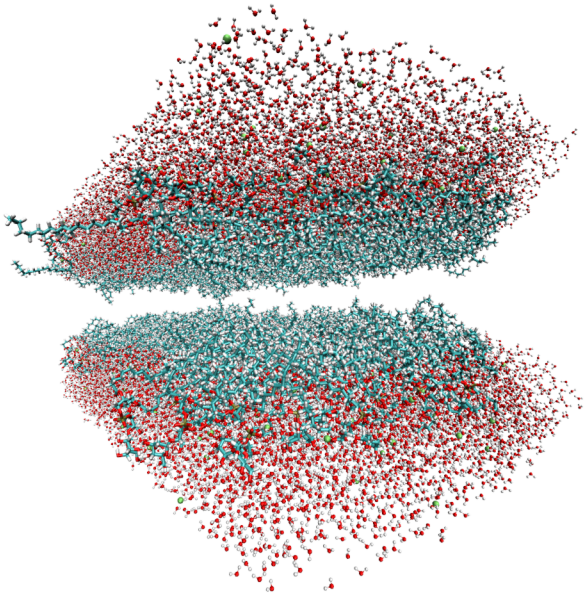


Figura 16: Finalização da primeira etapa de equilíbrio: NVT. Nota-se que os lipídios possuem restrição de movimento nos átomos de fósforo, mantendo a posição central deles no eixo Z durante essa etapa.

Após a fase NVT, o sistema entra na segunda fase de equilíbrio, desta vez sob o ensemble NPT (Número de partículas, Pressão e Temperatura constantes). Nesta etapa, além de manter a temperatura controlada, a pressão do sistema também é ajustada e estabilizada em torno do valor de referência (geralmente 1 bar), permitindo que a densidade do sistema se ajuste adequadamente. Isso é necessário para garantir que as dimensões da caixa de simulação e, consequentemente, a área por lipídio e a espessura da membrana, sejam representativas das condições experimentais ou do estado físico esperado para o sistema. A Figura 17 ilustra a configuração do sistema ao término desta fase de equilíbrio NPT. Conforme indicado na legenda da figura, as restrições de movimento no eixo Z para os lipídios foram mantidas também durante esta etapa. A manutenção dessas restrições continua sendo importante para evitar artefatos ou desorganização prematura da bicamada, garantindo uma transição suave para a produção por dinâmica molecular.

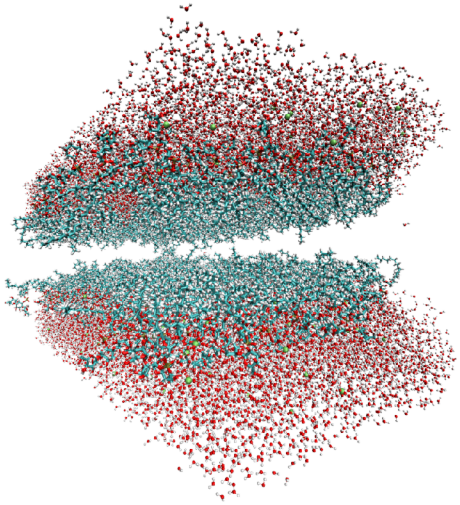


Figura 17: Finalização da segunda etapa de equilíbrio: NPT. Nota-se que os lipídios possuem restrição de movimento nos átomos de fósforo, mantendo a posição central deles no eixo Z durante essa etapa.

Após as etapas de equilíbrio NVT e NPT com restrições, o sistema foi finalmente submetido à simulação de dinâmica molecular. Nessa fase de produção, todas as restrições de movimento aplicadas aos lipídios foram removidas. Isso garante que o sistema se movimente livremente, e o comportamento dinâmico da bicamada lipídica pode ser observado e analisado. A Figura 18 apresenta uma visualização da estrutura do sistema após 100 ns de DM. Nesta imagem, pode-se observar a bicamada lipídica completamente móvel e flexível, exibindo flutuações e dinâmica características de um sistema de membrana em equilíbrio. É a partir desta trajetória resultante da dinâmica molecular que as propriedades de interesse (como área por lipídio, espessura da membrana, parâmetros de ordem, difusão, etc.) são tipicamente calculadas.

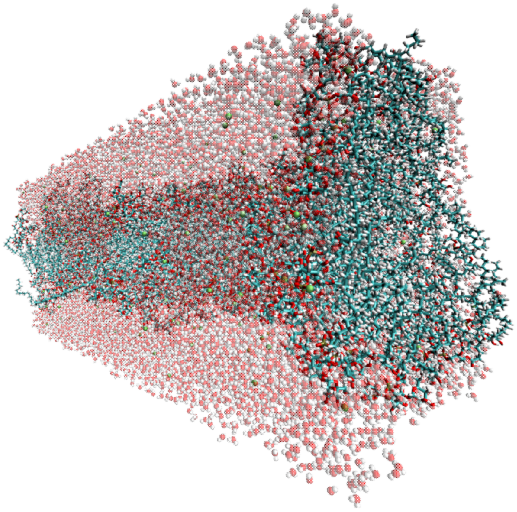
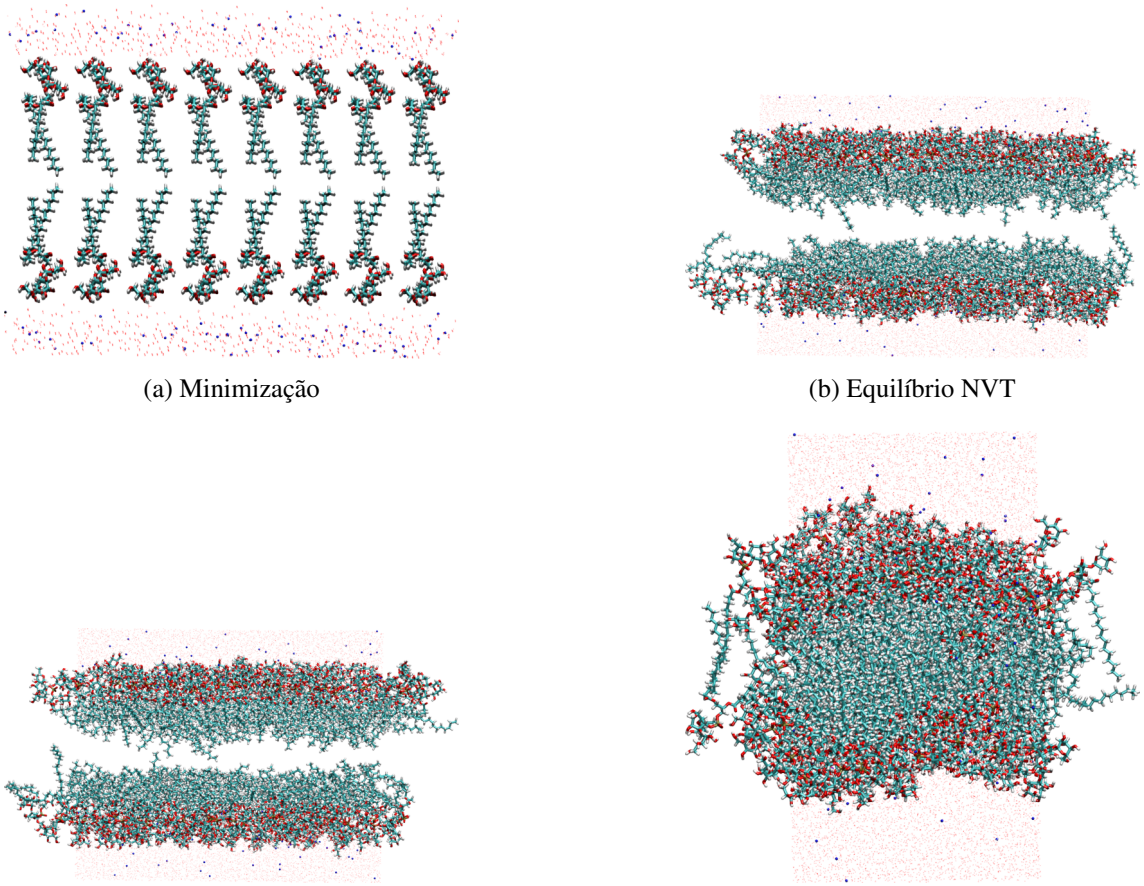


Figura 18: Estrutura final após a execução da dinâmica molecular, sem restrição de movimentos no sistema, permitindo a livre movimentação de todas as moléculas.

5.4.7 PIM_2

A análise de estabilidade do sistema PIM_2 puro é apresentada na Figura 20a. O RMSD estabiliza em 20 ns, indicando que o sistema atingiu o equilíbrio conformacional esperado de acordo com [51]. A análise de área por lipídio (Figura 20b) corrobora essa estabilidade. A evolução visual da bicamada ao longo das etapas da simulação é mostrada na Figura 19.



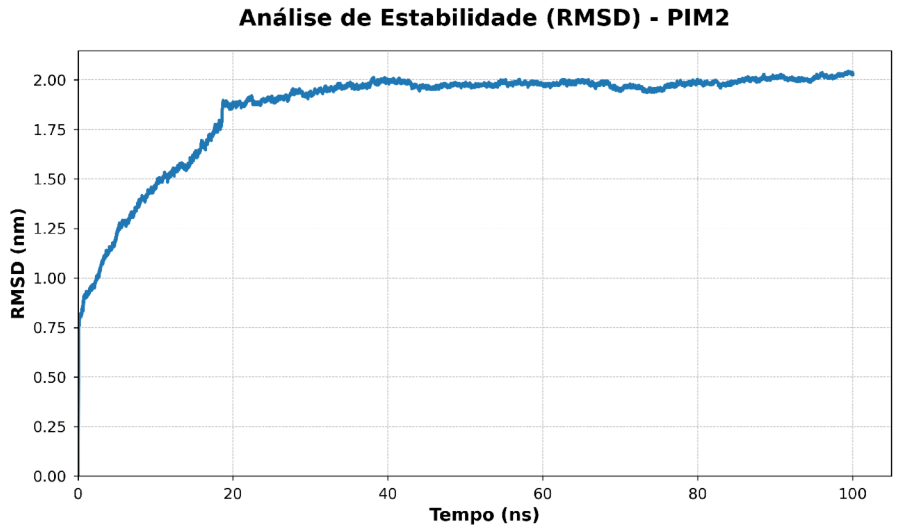
(a) Minimização

(b) Equilíbrio NVT

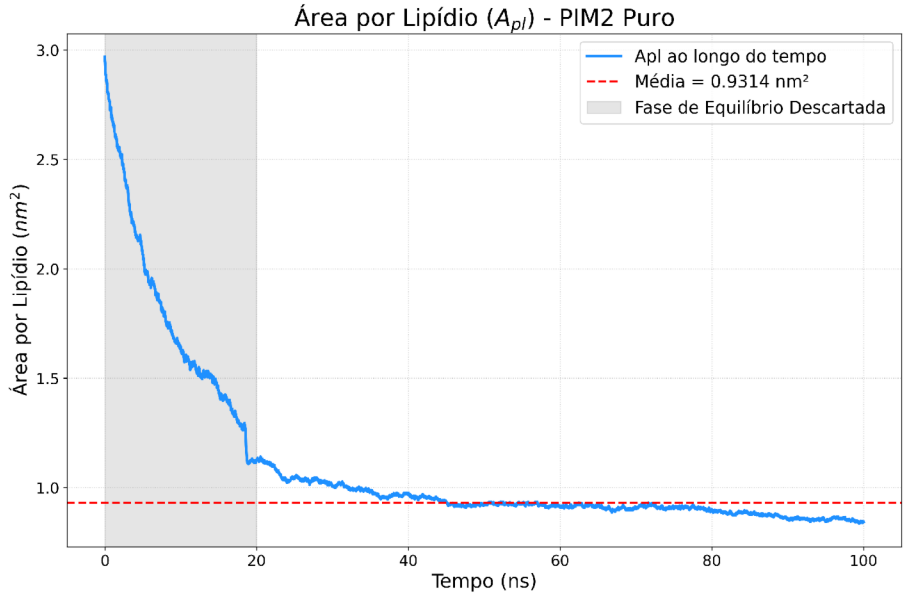
(c) Equilíbrio NPT

(d) Produção (DM)

Figura 19: Evolução do sistema PIM_2 puro: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM).



(a) Análise de RMSD.

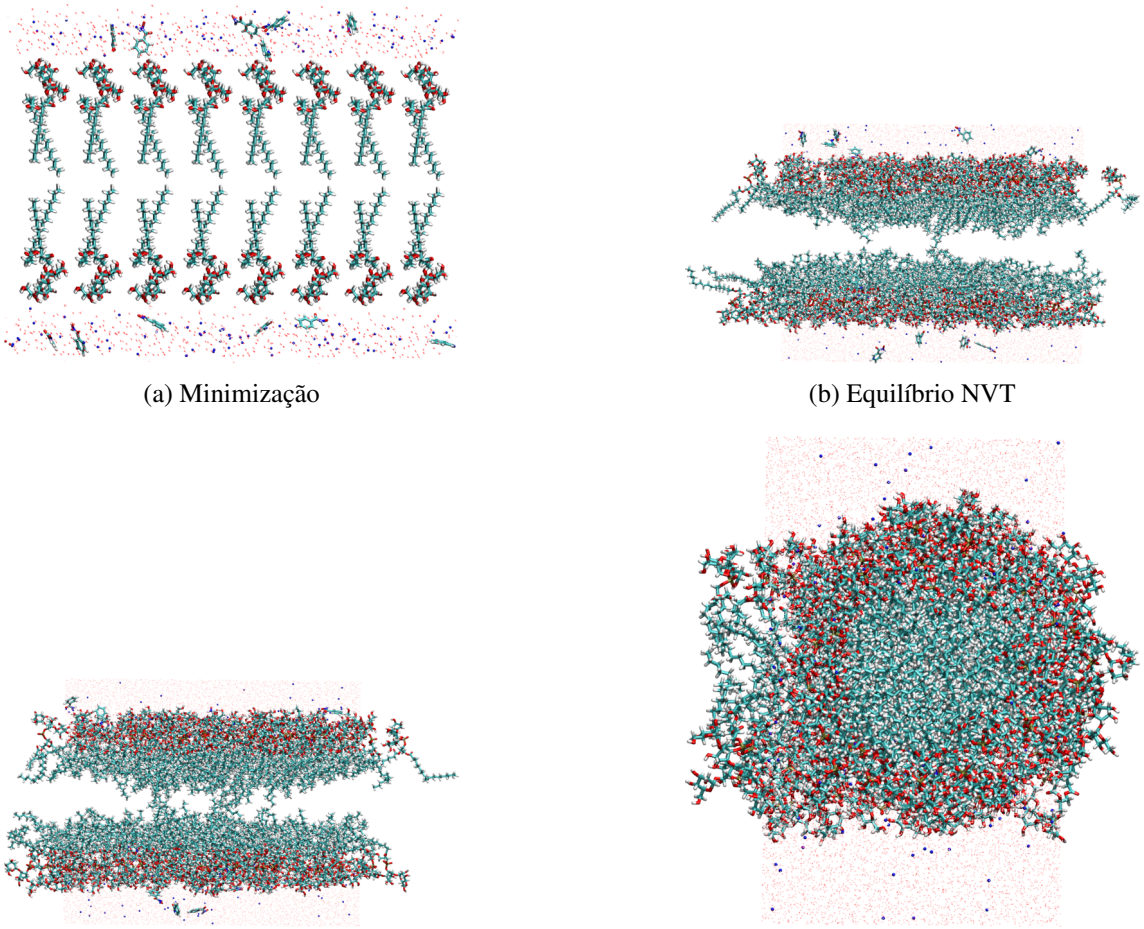


(b) Análise da área por lipídio (A_{pl}).

Figura 20: Gráficos de análise para o sistema PIM_2 puro.

5.4.8 PIM_2 com Benzohidroxamato

Para o sistema contendo o fármaco Benzohidroxamato (BH), o RMSD (Figura 22a) demonstra a estabilização da bicamada, em concordância com [51]. A análise de área por lipídio é mostrada na Figura 22b, e a evolução visual do sistema é apresentada na Figura 21.



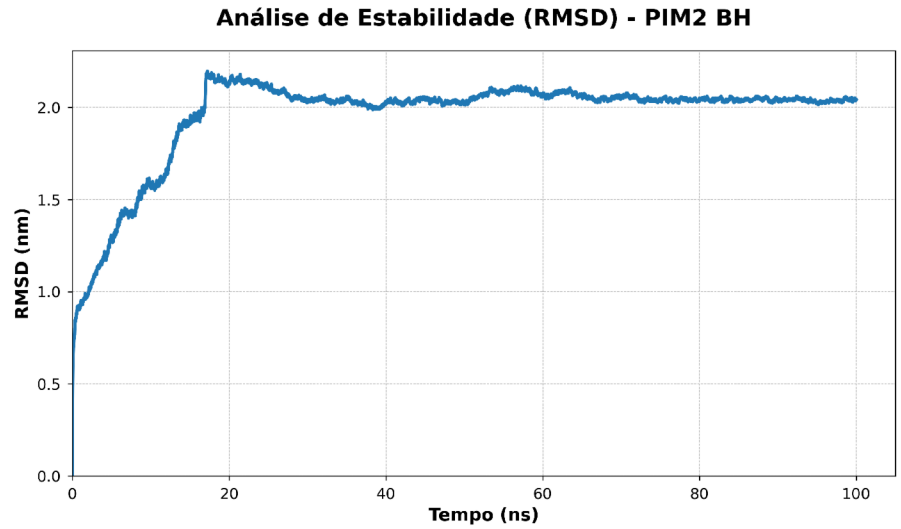
(a) Minimização

(b) Equilíbrio NVT

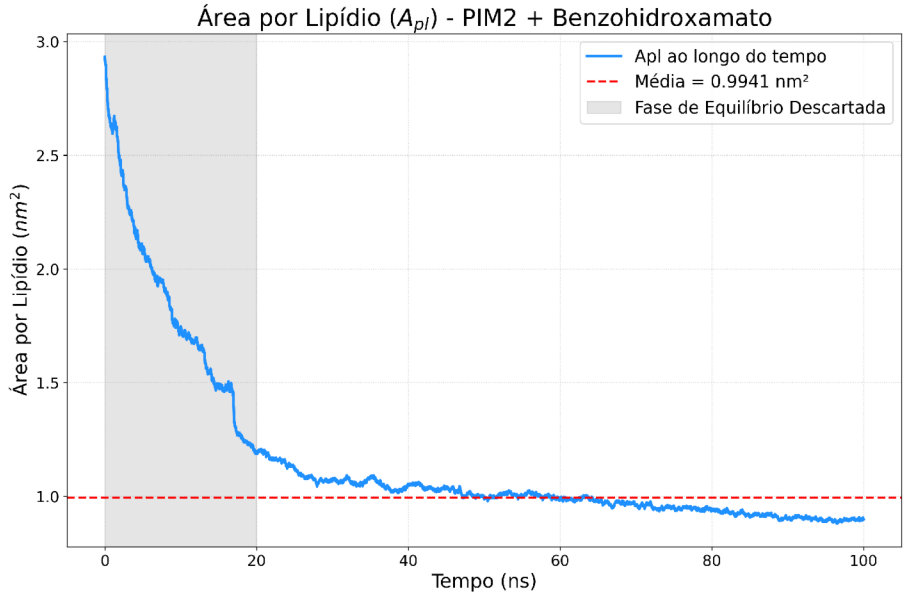
(c) Equilíbrio NPT

(d) Produção (DM)

Figura 21: Evolução do sistema PIM_2 com Benzohidroxamato: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM).



(a) Análise de RMSD.

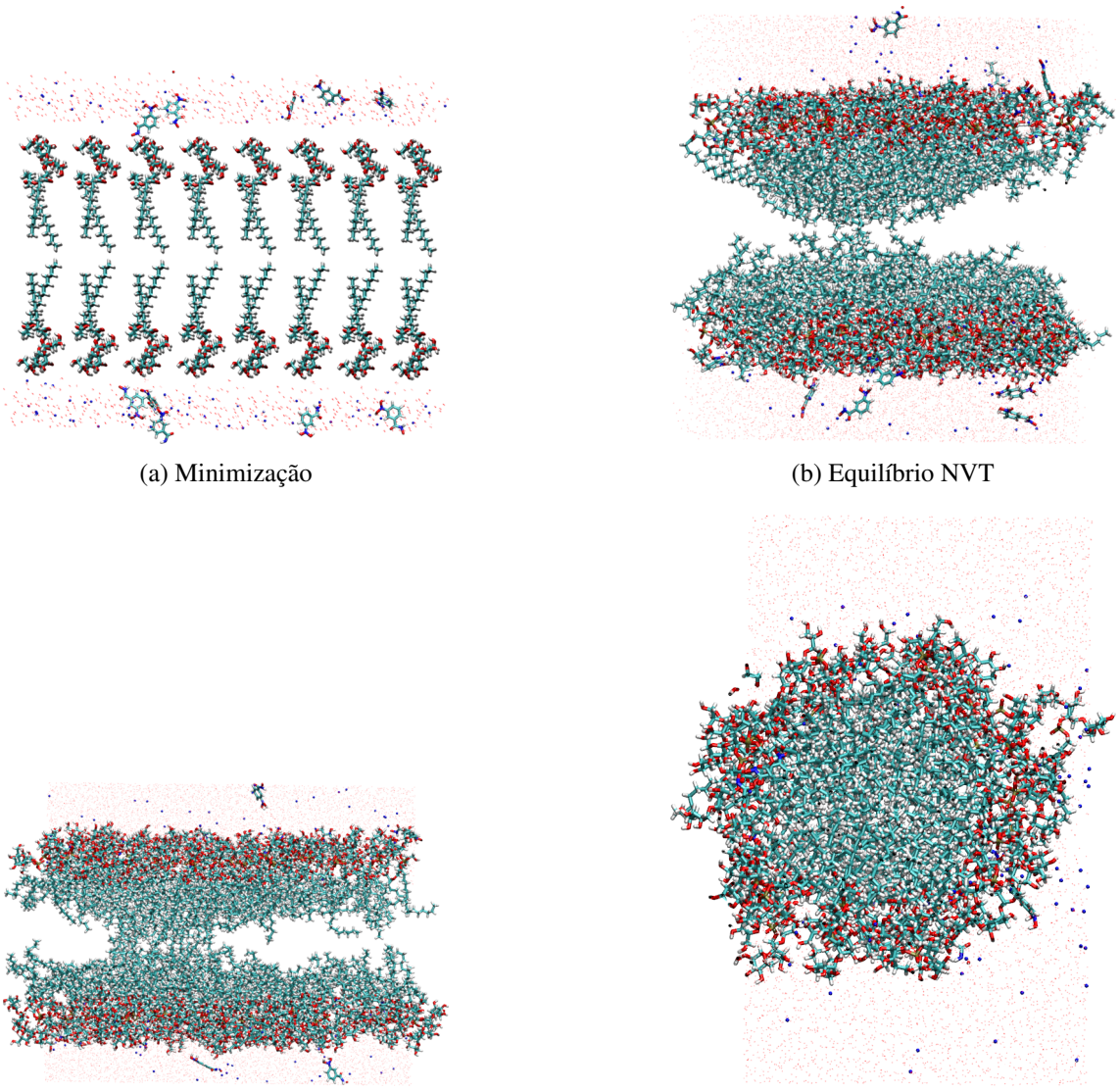


(b) Análise da área por lipídio (A_{pl}).

Figura 22: Gráficos de análise para o sistema PIM_2 com Benzohidroxamato.

5.4.9 PIM_2 com Nitrobenzohidroxamato

A estabilidade do sistema PIM_2 com nitrobenzohidroxamato (NBH) é confirmada pela análise de RMSD, que em torno 40 ns o sistema estabiliza e não possui flutuações elevadas de RMSD, variando menos de 0.1nm por ns. (Figura 24a) E pela área por lipídio (Figura 24b) concomitante com os outros dois sistemas PIM_2. A evolução visual do sistema é mostrada na Figura 23.



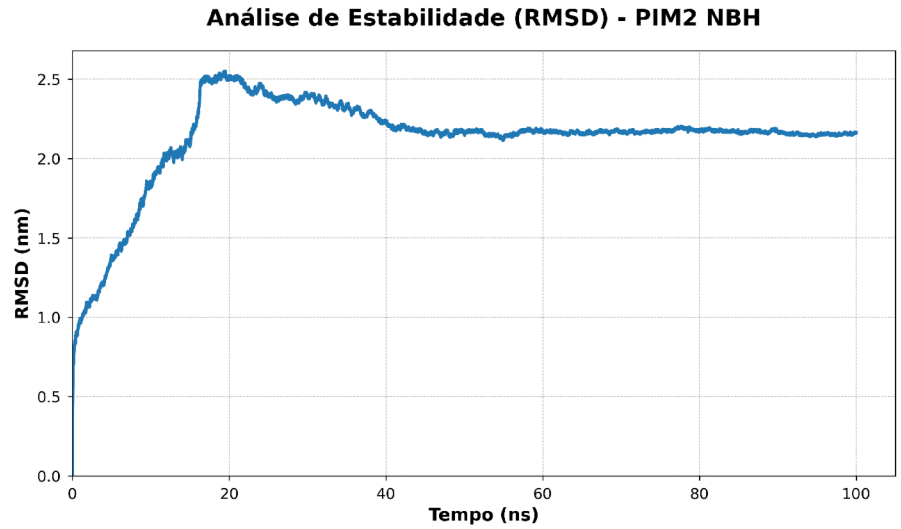
(a) Minimização

(b) Equilíbrio NVT

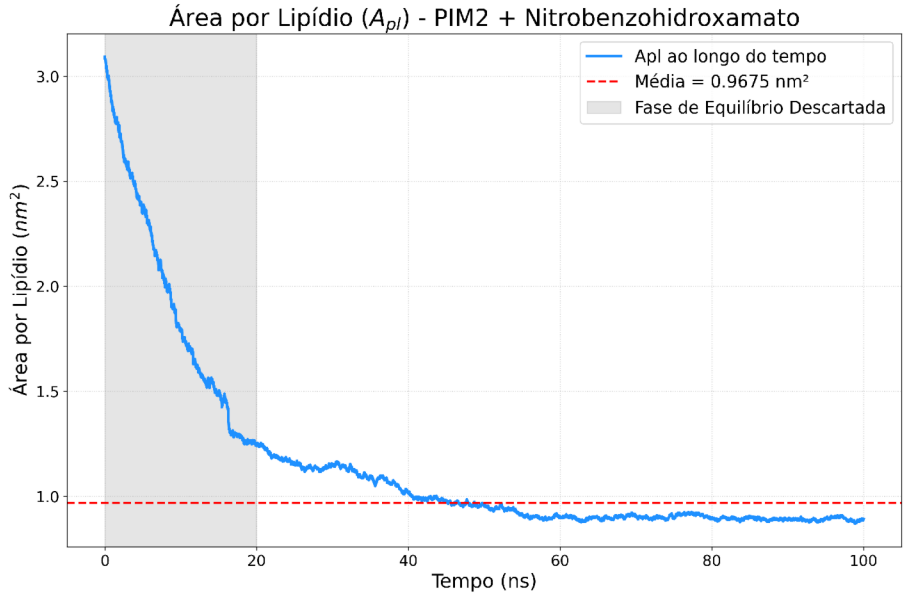
(c) Equilíbrio NPT

(d) Produção (DM)

Figura 23: Evolução do sistema PIM_2 com Nitrobenzohidroxamato: (a) Minimização, (b) NVT, (c) NPT, (d) Produção (DM).



(a) Análise de RMSD.



(b) Análise da área por lipídio (A_{pl}).

Figura 24: Gráficos de análise para o sistema PIM_2 com Nitrobenzohidroxamato.