Alexandre Barbosa de Almeida

Predição de Estrutura Terciária de Proteínas com Técnicas Multiobjetivo no Algoritmo de Monte Carlo

Goiânia
2016

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR AS TESES E DISSERTAÇÕES ELETRÔNICAS NA BIBLIOTECA DIGITAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio da Biblioteca Digital de Teses e Dissertações (BDTD/UFG), regulamentada pela Resolução CEPEC nº 832/2007, sem ressarcimento dos direitos autorais, de acordo com a Lei nº 9610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou *download*, a título de divulgação da produção científica brasileira, a partir desta data.

1. Identificação do material bibliográfico: **[X] Dissertação** **[ ] Tese**

2. Identificação da Tese ou Dissertação

Nome completo do autor: ALEXANDRE BARBOSA DE ALMEIDA

Título do trabalho: PREDIÇÃO DE ESTRUTURA TERCIÁRIA DE PROTEÍNAS COM TÉCNICAS MULTIOBJETIVO NO ALGORITMO DE MONTE CARLO

3. Informações de acesso ao documento:

Concorda com a liberação total do documento [X] SIM [ ] NÃO[1]

Havendo concordância com a disponibilização eletrônica, torna-se imprescindível o envio do(s) arquivo(s) em formato digital PDF da tese ou dissertação.

Alexandre Barbosa de Almeida

Assinatura do (a) autor (a)

Data: 05 / 08 / 2016

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. A extensão deste prazo suscita justificativa junto à coordenação do curso. Os dados do documento não serão disponibilizados durante o período de embargo.

Alexandre Barbosa de Almeida

Predição de Estrutura Terciária de Proteínas com Técnicas Multiobjetivo no Algoritmo de Monte Carlo

Dissertação apresentada ao Programa de Pós–Graduação do Instituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Ciência da Computação

Orientadora: Prof[a]. Dra. Telma W. L. Soares

Coorientador: Prof. Dr. Rodrigo Antonio Faccioli

Goiânia
2016

Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Barbosa de Almeida, Alexandre
Predição de Estrutura Terciária de Proteínas com Técnicas Multiobjetivo no Algoritmo de Monte Carlo [manuscrito] / Alexandre Barbosa de Almeida. - 2016.
129 f.: il.

Orientador: Profa. Dra. Telma Woerle de Lima Soares; co orientador Dr. Rodrigo Antonio Faccioli.
Dissertação (Mestrado) - Universidade Federal de Goiás, Instituto de Informática (INF), Programa de Pós-Graduação em Ciência da Computação, Goiânia, 2016.
Bibliografia.
Inclui siglas, abreviaturas, lista de figuras, lista de tabelas.

1. Predição da Estrutura Terciária de Proteínas. 2. Otimização Multiobjetivo. 3. Monte Carlo Metropolis. 4. Monte Carlo com Dominância. I. Woerle de Lima Soares, Telma, orient. II. Título.

CDU 004

Alexandre Barbosa de Almeida

Predição de Estrutura Terciária de Proteínas com Técnicas Multiobjetivo no Algoritmo de Monte Carlo

Dissertação defendida no Programa de Pós-Graduação em Ciência da Computação do Instituto de Informática da Universidade Federal de Goiás como requisito parcial para obtenção do título de Mestre em Ciência da Computação, aprovada em 17 de junho de 2016, pela Banca Examinadora constituída pelos professores:

Profa. Dra. Telma Woerle de Lima Soares
Instituto de Informática - UFG
Presidente da Banca

Prof. Dr. Rodrigo Antonio Faccioli
Coorientador
Ciência da Computação – Centro Universitário Barão de Mauá/SP

Prof. Dr. Wellington Santos Martins
Instituto de Informática – UFG

Prof. Dr. Salviano de Araújo Leão

Por razões e fatos indubitáveis, além de sentimentos de gratidão inexprimíveis, dedico este mestrado inteiramente a Deus! Há um propósito que delineou o curso de minha vida até este momento e que me impele a continuar acreditando, pela fé, que não existe um término, mas uma continuação, onde os cenários se alteram e a vida se esforça para encontrar o caminho escrito por Ele.

Agradecimentos

Não há outra forma de iniciar os agradecimentos senão homenageando os meus pais e toda a minha família. De modo sucinto, porém com intenso carinho, lhes digo: obrigado por absolutamente tudo!

Às amizades que encontrei e pensei serem passageiras, mas ficaram e se tornaram uma segunda família, agradeço imensamente, sem o apoio de vocês este trabalho não seria possível (mentira, seria sim!): à Karen Cristina, agradeço por ter tido uma importância sem igual durante o período deste meu mestrado; ao Luiz Eduardo, Pedro Paulo, Idney Resplandes, Letícia de Sá e Ariane Bitencourt, vocês são referências para mim de caráter, lealdade, força, determinação e fé!

Agradecimentos especiais à Wanessa Carvalho, pela amizade dedicada e pela contribuição com a revisão dos conceitos biológicos. Obrigado Ana e Wagner Bandeira, grato pela oportunidade, compreensão e apoio! Estendo esse mesmo obrigado, seguido de um forte abraço, ao Amilton Rogério (comandante!), à Janinne Barcelos (uma mulher de *classe!*) e à Rhanna Asevedo (as tapiocas!), grato pelas conversas e tardes de almoço. Aos amigos do Instituto de Física e Instituto de Informática (INF) da Universidade Federal de Goiás (UFG), não é possível listar todos os nomes, mas obrigado pelo apoio, sei o quanto torceram por mim. Ao grupo de pesquisa em computação evolucionária do INF/UFG e, em especial, à Michelle Duarte, você foi fundamental nessa conquista!

À minha orientadora Profa. Dra. Telma Soares e ao Prof. Dr. Anderson Soares, agradeço por ter acolhido um físico neste grupo de computeiros, além da oportunidade de trabalhar com dois dos melhores professores do Instituto de Informática, é admirável a dedicação, o comprometimento e o respeito que dispensam aos seus alunos. Ao meu coorientador de São Paulo, Prof. Dr. Rodrigo Faccioli, agradeço a paciência e por todos os ensinamentos transmitidos em nossas inúmeras reuniões por Skype e e-mails trocados, apesar de nunca termos nos conhecido pessoalmente, não vou esquecer que ainda devo aquela picanha tão comentada e o bom e velho *pequi do Goiás*. Por fim, ao Prof. Dr. Salviano Leão do Instituto de Física, pois foi lá que tudo começou. Olha que jornada, professor! E pensar que ainda é só o começo, no entanto, os 10% ficam cada vez mais próximos.

Obrigado!

"Tenha coragem de seguir o que seu coração e intuição dizem. Eles já sabem o que você realmente deseja. Todo resto é secundário."

– Steve Jobs

Resumo

ALMEIDA, A.B. **Predição de Estrutura Terciária de Proteínas com Técnicas Multiobjetivo no Algoritmo de Monte Carlo**. Goiânia, 2016. 129 p. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, Universidade Federal de Goiás.

As proteínas são vitais para as funções biológicas de todos os seres na Terra. Entretanto, somente apresentam função biológica ativa quando encontram-se em sua estrutura nativa, que é o seu estado de mínima energia. Portanto, a funcionalidade de uma proteína depende, quase que exclusivamente, do tamanho e da forma de sua conformação nativa. Porém, de todas as proteínas conhecidas no mundo, menos de 1% tem a sua estrutura resolvida. Deste modo, vários métodos de determinação de estruturas de proteínas têm sido propostos, tanto para experimentos *in vitro* quanto *in silico*. Este trabalho propõe um novo método *in silico* denominado Monte Carlo com Dominância, o qual aborda o problema da predição de estrutura de proteínas sob o ponto de vista *ab initio* e de otimização multiobjetivo, considerando, simultaneamente, os aspectos energéticos e estruturais da proteína. Para o tratamento *ab initio* utiliza-se o *software* GROMACS para executar as simulações de Dinâmica Molecular, enquanto que para o problema da otimização multiobjetivo emprega-se o *framework* ProtPred-GROMACS (2PG), o qual utiliza algoritmos genéticos como técnica de soluções heurísticas. O Monte Carlo com Dominância, nesse sentido, é como uma variante do tradicional método de Monte Carlo Metropolis. Assim, o objetivo é o de verificar se a predição da estrutura terciária de proteínas é aprimorada levando-se em conta também os aspectos estruturais. O critério energético de Metropolis e os critérios energéticos e estruturais da Dominância foram comparados empregando o cálculo de RMSD entre as estruturas preditas e as nativas. Foi verificado que o método de Monte Carlo com Dominância obteve melhores soluções para duas de três proteínas analisadas, chegando a cerca de 53% de diferença da predição por Metropolis.

Palavras - chave: Predição da Estrutura Terciária de Proteínas. Otimização Multiobjetivo. Monte Carlo Metropolis. Monte Carlo com Dominância.

Abstract

ALMEIDA, A.B. **Proteins Tertiary Structure Prediction with Multiobjective Techniques in Monte Carlo Algorithm**. Goiânia, 2016. 129 p. Master Thesis. Informatics Institute, Federal University of Goiás.

Proteins are vital for the biological functions of all living beings on Earth. However, they only have an active biological function in their native structure, which is a state of minimum energy. Therefore, protein functionality depends almost exclusively on the size and shape of its native conformation. However, less than 1% of all known proteins in the world has its structure solved. In this way, various methods for determining protein structures have been proposed, either *in vitro* or *in silico* experiments. This work proposes a new *in silico* method called Monte Carlo with Dominance, which addresses the problem of protein structure prediction from the point of view of *ab initio* and multi-objective optimization, considering both protein energetic and structural aspects. The software GROMACS was used for the *ab initio* treatment to perform Molecular Dynamics simulations, while the framework ProtPred-GROMACS (2PG) was used for the multi-objective optimization problem, employing genetic algorithms techniques as heuristic solutions. Monte Carlo with Dominance, in this sense, is like a variant of the traditional Monte Carlo Metropolis method. The aim is to check if protein tertiary structure prediction is improved when structural aspects are taken into account. The energy criterion of Metropolis and energy and structural criteria of Dominance were compared using RMSD calculation between the predicted and native structures. It was found that Monte Carlo with Dominance obtained better solutions for two of three proteins analyzed, reaching a difference about 53% in relation to the prediction by Metropolis.

Keywords: Protein Tertiary Structure Prediciton. Multi-objective Optimization. Monte Carlo Metropolis. Monte Carlo with Dominance.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

UFG	Universidade Federal de Goiás
INF	Instituto de Informática
PSP	*Protein Structure Prediction*
GROMACS	GROningen MAchine for Chemical Simulations
2PG	ProtPred-GROMACS
AE	Algoritmo Evolutivo
AG	Algoritmo Genético
POMO	Problema de Otimização Multiobjetivo
MOEA	*Multi-Objective Evolutionary Algorithm*
RMSD	*Root-Mean-Square Deviation*
aSASA	*apolar Solvent-Accessible Surface Area*
pSASA	*polar Solvent-Accessible Surface Area*
GBSA	*Generalized Born Superficie Area*
RG	Raio de Giro

Sumário

CAPÍTULO 1

Introdução

"As espécies evoluem pelo princípio da seleção natural e a sobrevivência do mais apto."

– Charles Darwin (A Teoria da Evolução, 1859)

As proteínas estão presentes em todos os seres vivos, executando funções diversas para a criação e a manutenção da vida na Terra (CARTER; WOLFENDEN, 2015). As proteínas são sintetizadas pelos ribossomos localizados no interior das células, onde existe um maquinário de síntese proteica equipado com RNAs transportadores (tRNAs) que levam os aminoácidos dispersos no citoplasma até o ribossomo. Então, quando o anticódon do tRNA encontra o seu complemento no códon da fita do RNA mensageiro (mRNA), o aminoácido se desprende do tRNA e inicia-se a formação de uma cadeia linear de aminoácidos, dando origem a um polipeptídeo que resultará na criação final de uma proteína (NELSON; COX, 2008).

Este trabalho inicia-se a partir desta sequência linear de aminoácidos, também conhecida como estrutura primária da proteína. Desde os trabalhos de Anfinsen (1973), existe a premissa de que a sequência de aminoácidos de uma cadeia polipeptídica contém todas as informações necessárias sobre como a proteína assume a sua estrutura tridimensional final, também conhecida como estrutura nativa. O estado nativo é a conformação espacial na qual a proteína desempenha uma função biológica ativa no organismo.

1.1 A Importância Estrutural das Proteínas

A **conformação** de uma proteína é o arranjo espacial formado pelos átomos que a constitui, ou seja, é a sua forma tridimensional no espaço. Vários fatores contribuem para a existência de diferentes conformações, entretanto, a natureza sempre tende a assumir aquelas termodinamicamente estáveis. Do ponto de vista físico, isto significa dizer que a energia livre de Gibbs (G) da proteína deve ser mínima. Deste modo, define-se a **estrutura**

nativa como sendo a conformação enovelada de uma proteína em seu estado de mínima energia livre e que desempenha alguma função biológica ativa. A conformação nativa da proteína é o estado mais estável de configuração espacial, onde pequenas mudanças no ambiente que a cerca pode ensejar alterações estruturais que podem afetar a sua função biológica.

Com exceção da água, as proteína são as moléculas mais abundantes do corpo humano. As proteínas realizam ações enzimáticas como catalisadoras de reações químicas, agem como componentes estruturais conferindo rigidez (proteínas fribrosas), outras são sinalizadoras extracelulares como a insulina, transmitindo sinais para tecidos distantes, ou são proteínas ligantes que transportam biomoléculas para diferentes locais no corpo, ou seja, são vitais para praticamente todas as funções orgânicas. A execução de todas estas atividades depende, exclusivamente, da proteína ter função biológica ativa, que por sua vez depende do seu estado nativo. Assim, conhecer o processo de formação da estrutura nativa é de assaz importância. Este processo de formação estrutural é conhecido como **enovelamento**, ou ***folding* de proteínas** (DILL et al., 2008).

A má formação do *folding* pode ocasionar várias desordens genéticas, causando ou contribuindo para o surgimento de muitas doenças, como o diabetes do tipo 2, Alzheimer, Parkinson, entre outros. Cerca de um quarto ou mais de todos os polipeptídeos sintetizados podem ser destruídos devido ao *folding* incorreto durante a sua formação. As doenças resultantes do *folding* incorreto de proteínas são chamadas de *amiloidoses*, elas ocorrem quando proteínas que são normalmente solúveis em água, mas ao serem secretadas da célula em um estado de *folding* incorreto, passam a ser insolúveis no meio extracelular, sendo convertidas em tipos de fibras denominadas *amilóides*, que se acumulam em tecidos ou órgãos alterando as suas funções naturais (NELSON; COX, 2008). Desta forma, investigar o mecanismo de *folding* desde os seus princípios físicos, ou ***ab initio***, pode contribuir para um entendimento mais preciso sobre o processo de enovelamento da proteína, o que pode auxiliar, inclusive, em uma compreensão mais ampla acerca da origem destas doenças.

1.2 Motivação e Justificativa

De acordo com as estatísticas referentes aos bancos de dados UniProt (2015) e RCSB Protein Data Bank (2015a), na publicação de 13 de abril de 2016, o UniProtKB/-TrEMBL contabilizou 63.686.057 sequências de proteínas depositadas, enquanto que o PDB registrou 109.850 estruturas de proteínas resolvidas, ou seja, apenas cerca de $0,17\%$ de todas as proteínas conhecidas (no mundo) tem a sua estrutura determinada. Este cenário faz com que exista uma alta demanda por pesquisa de métodos de determinação de estruturas de proteínas, no que ficou conhecido como o problema da predição de estrutura de proteínas, ou simplesmente, o problema do PSP (do inglês, *Protein Structure Prediction).*

Este trabalho emprega métodos *in silico* para tentar determinar, em especial, a estrutura terciária de proteínas, ou seja, por meio de técnicas de simulações computacionais objetiva-se predizer a forma estrutural de uma proteína, muitas vezes referida como **proteína-alvo**. Todavia, em geral, a determinação computacional não é tão precisa quanto os métodos de bancada de laboratório, a exemplo dos experimentos de cristalografia com difração de raios X e os de ressonância magnética nuclear (RMN). Deste modo, o que de fato ocorre é uma série de tentativas de predição envolvendo as mais diversas abordagens que podem ser separadas em três categorias: modelagem por homologia, *threading* e *ab initio* (ECHENIQUE, 2007). Este trabalho faz uso da abordagem ***ab initio***, ou primeiros princípios, que considera os princípios físicos envolvidos durante o processo de *folding*.

A justifica de se utilizar métodos *in silico* é por este ser muito mais barato e demandar muito menos tempo que os experimentos de bancada. Em geral, os experimentos de bancada exigem equipamentos caros e tempo para preparação de amostras, como no caso da cristalização de proteínas para a difração de raios X, sendo que nem todas as proteínas podem formar estruturas cristalinas (DRENTH, 1994). Entretanto, Cheng et al. (2015) desenvolveram uma nova técnica promissora de determinação de estrutura de proteínas e de complexos macromoleculares denominada *Cryo-electron Microscopy* (cryo-EM) que não necessita de cristalização, produzindo imagens de qualidade sem precedentes, permitindo que as estruturas possam ser determinadas com resolução quase a nível atômico. Mas, em comparação ao método de cristalografia por difração de raios X, ainda é uma técnica que precisa ser aprimorada.

Embora as estruturas preditas *in silico* não sejam tão exatas quanto as de bancada de laboratório, geralmente podem fornecer informações muito valiosas acerca do mecanismo de *folding*, algo ainda não completamente compreendido (DILL et al., 2008). Os experimentos *in silico* permitem avaliar mecanismos para elaboração de um modelo de predição cada vez mais acurado, sendo que a qualidade de uma predição depende da semelhança estrutural (similaridade) entre a proteína predita e a proteína-alvo, onde a estrutura da proteína-alvo já tenha sido resolvida por algum tipo de experimento de bancada (por exemplo, RMN ou cristalografia).

Existem várias propostas de predição para o enovelamento, seja comparando com outras proteínas similares (homologia) a fim de encontrar padrões estruturais, ou realizando buscas estatísticas de alinhamento com outras proteínas de um banco de dados (*threading*), ou começando do zero, desde o início, ao tentar modelar os princípios físicos envolvidos no processo de *folding* (*ab initio*). Apesar de que muitas proteínas enovelam-se espontaneamente, de forma independente, sujeitas apenas às condições ambientes, outras já necessitam de serem assistidas por outras proteínas chamadas de *chaperonas*. Então, diante de tal complexidade relativa ao mecanismo de *folding*, abordar este problema também do ponto de vista de otimização parece uma alternativa promissora.

1.3 Metodologia do Trabalho

A abordagem *ab initio* utiliza princípios físicos para realizar a predição de estruturas de proteínas (LEE; WU; ZHANG, 2015). Logo, simulações de **Dinâmica Molecular** são essenciais neste tipo de problema. Existem *frameworks* na literatura que realizam cálculos de modelagem molecular, como por exemplo o GROMACS (HESS et al., 2008), o TINKER (PONDER, 2001), o FAUNUS (LUND; TRULSSON; PERSSON, 2008), entre outros. O GROMACS (2015a) tem se destacado por apresentar acurácia e rapidez nos cálculos das propriedades físicas de proteínas, além de ser um *software open-source*. Em geral, os algoritmos de simulação de Dinâmica Molecular modelam a proteína sujeitas a um campo de força e imersas em algum tipo de solvente.

Para a predição das estruturas terciárias foi utilizado o *framework* ProtPred-GROMACS (2PG). O 2PG permite modelar o PSP como um problema de otimização, aplicando **algoritmos evolutivos** (AEs) para a **otimização multiobjetivo** e utiliza o GROMACS para os cálculos das propriedades físicas das proteínas (FACCIOLI, 2012).

Em teoria de otimizações, normalmente é necessário maximizar ou minimizar uma dada função, usualmente chamada de **função objetivo** (BHATTI, 2000). Dependendo do problema, será necessário otimizar apenas um objetivo (mono-objetivo) ou vários objetivos (**multiobjetivo**) simultaneamente. Este tipo de abordagem é aplicado a problemas complexos que não apresentam soluções analíticas triviais ou que são de difícil modelagem, seja ela física ou puramente matemática. Uma das técnicas de se resolver problemas de otimização multiobjetivo e que vem demostrando ótimos resultados são os algoritmos genéticos, ou evolutivos (DEB, 2001; DEJONG, 2006).

No caso em que a abordagem multiobjetivo seja para minimizar a função objetivo, as soluções obtidas são denominadas **soluções factíveis**. O conjunto das soluções factíveis é chamado de **espaço de busca**. É preciso percorrer o espaço de busca para **tomar uma decisão** de qual solução é melhor do que a outra, então para isto aplica-se um critério denominado **dominância** sobre todas as soluções factíveis. As melhores soluções são aquelas que passam pelo critério de dominância, tais soluções são ditas **soluções eficientes** e o conjunto das soluções eficientes é denominado de **fronteira de Pareto**.

No que diz respeito aos algoritmos de predição, em geral, a conformação de uma proteína é modelada por meio de três ângulos denominados **ângulos diedros** ϕ, ψ e χ (lê-se "phi", "psi" e "chi", respectivamente). É usual na literatura que estes algoritmos mantenham fixos certos parâmetros, tais como os comprimentos de ligação e ângulos de ligação entre os átomos da proteína, enquanto que novos valores para os ângulos diedros são computados, originando novas conformações. Deste modo, os ângulos diedros são comumente chamados de **parâmetros livres**, conferindo apenas três graus de liberdade rotacionais à conformação.

Este trabalho emprega o 2PG para o problema do PSP sob a perspectiva *ab initio* e de otimização. O 2PG provê algoritmos evolutivos que irão alterar os valores dos parâmetros livres da proteína originando novas conformações estruturais. O conjunto destas novas soluções formam o que é chamado de **espaço de busca**. Entretanto, somente serão aceitas conformações que satisfaçam a um dado critério energético e/ou estrutural (função objetivo). Para predizer a estrutura nativa é preciso buscar por soluções que miniminizem a função objetivo. Para isto, dois critérios de aceitação são empregados: o tradicional método de **Monte Carlo Metropolis** (mono-objetivo) e o novo método proposto por este trabalho, o **Monte Carlo com Dominância** (multiobjetivo).

O Monte Carlo Metropolis faz uso de parâmetros específicos do campo de força para a simulação de Dinâmica Molecular, porém, não se sabe exatamente qual tipo de campo de força é melhor para se trabalhar com proteínas. O que se pretende com este trabalho é olhar para o espaço de busca sem estar preso aos parâmetros do campo de força por meiro do método de Monte Carlo com Dominância, o qual será incorporado ao 2PG. Enquanto que os algoritmos de Dinâmica Molecular realizam uma busca local, com este novo método pretende-se ter uma **busca global no espaço de busca**. A hipótese é de que desta maneira haverá uma melhor exploração do espaço de busca, permitindo avaliar vários outros objetivos a fim de melhorar a predição.

Os multiobjetivos a serem tratados neste trabalho são:

1. **Energia potencial**: soma das energias das interações covalentes e não-covalentes;
2. **Área hidrofóbica**: superfície de acessibilidade do solvente aos aminoácidos apolares;
3. **Área hidrofílica**: superfície de acessibilidade do solvente aos aminoácidos polares;
4. **Energia de solvatação**: emprega o método GBSA (*Generalized Born Surface Area*) para calcular energia livre de solvatação considerando a superfície de acessibilidade de um modelo de solvente implícito;
5. **Raio de giro**: auxilia a determinar o estado de enovelamento da proteína.

Estes objetivos, ou multiobjetivos, serão analisados simultaneamente empregando o conceito de dominância no algoritmo de Monte Carlo, algo até então inédito na literatura de predição de proteínas.

A dominância será aplicada sobre duas soluções, nomeadas de solução atual e solução nova. Inicialmente estas duas soluções são iguais e, a seguir, são aplicadas mudanças aleatórias (requisito do algoritmo de Monte Carlo) nos parâmetros livres (espaço amostral) da solução nova, obtendo-se uma nova conformação estrutural. Neste ponto é

verificada a dominância como o critério de Monte Carlo: caso esta solução nova domine a solução corrente, ela será aceita e solução atual recebe a solução nova, caso contrário, a estrutura aceita continua sendo a da solução atual. Isto se repete iterativamente de acordo com o número de passos (*steps*) de Monte Carlo.

Como normalmente os passos podem ser números suficientemente grandes, define-se um parâmetro chamado de *frequência de Monte Carlo*, o qual determina a frequência com que serão salvas essas estruturas. Uma vez obtido esse conjunto de soluções preditas, estas serão avaliadas de acordo com o cálculo de RMSD em relação à proteína-alvo (MAIOROV; CRIPPEN, 1994). Para este trabalho serão avaliadas três proteínas-alvo: 1VII, 1FSD e 1LE0.

Quanto menor o RMSD, mais semelhante a estrutura da proteína predita estará da estrutura da proteína-alvo (estrutura nativa). O mesmo é feito para as estruturas preditas pelo critério de Metropolis, onde o objetivo será sempre energia potencial. Como será visto, a energia potencial depende dos parâmetros dos potenciais do campo de força adotado para a simulação. A energia potencial consiste de um somatório das contribuições energéticas das ligações covalentes e das não-covalentes. Todavia, os critérios estruturais (não energéticos) não dependem do campo de força.

Por fim, é feita uma comparação entre os RMSDs obtidos pela técnica de Dominância com a de Metrópolis no algoritmo de Monte Carlo. O objetivo é o de verificar se a análise multiobjetivo contribuiu com melhores predições ao conseguir explorar mais o espaço de busca das soluções.

1.4 Objetivos

A implementação deste novo método de Monte Carlo com Dominância avaliará, simultaneamente, os seguintes critérios:

a) Energia potencial e energia de solvatação;

b) Energia potencial e área hidrofóbica;

c) Área hidrofóbica e área hidrofílica;

d) Raio de giro e área hidrofílica;

e) Raio de giro e energia de solvatação.

Deste modo, a hipótese a ser confirmada é de que os valores de RMSD obtidos pelo método de Monte Carlo com Dominância sejam, pelo menos para uma dada função objetivo, menores que aqueles obtidos pelo Monte Carlo Metropolis, corroborando que a abordagem por dominância pode ser mais efetiva ao explorar de forma global o espaço de busca.

1.5 Organização do Trabalho

Este trabalho está organizado da seguinte maneira: o capítulo 2 discorre acerca dos conceitos biológicos necessários à contextualização do problema da predição de estruturas de proteínas, dando respaldo a um *entendimento biológico mínimo* para a modelagem computacional do PSP. O capítulo 3 trata da modelagem computacional para a predição estrutural de proteínas e o capítulo 4 ocupa-se em explicar o problema da otimização multiobjetivo.

Já os capítulos finais tratam dos aspectos computacionais e da proposta deste trabalho. O capítulo 5 explica em detalhes os *softwares* utilizados para a predição das proteínas, em especial, o ProtPred-GROMACS (2PG), que trata o PSP como um problema de otimização; e o GROMACS, responsável pela abordagem *ab initio* onde executa os cálculos de Dinâmica Molecular. O capítulo 6 expõe a teoria formal do método de Monte Carlo e a proposta inédita do Monte Carlo com Dominância. O capítulo 7 descreve os resultados obtidos e suas análises. E, por fim, o capítulo 8 é dedicado às conclusões obtidas deste trabalho.

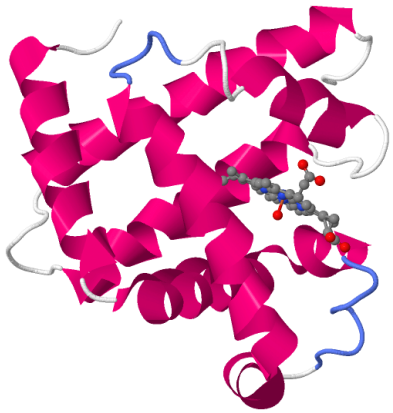
CAPÍTULO 2

PROTEÍNAS

O objetivo deste capítulo é o de apresentar um entendimento biológico mínimo necessário para compreender os aspectos computacionais envolvidos no que é conhecido como o problema da *Predição de Estrutura de Proteínas*, ou do inglês, PSP (*Protein Structure Prediction*). A exposição teórica deste capítulo, em sua maioria, baseia-se na obra de Nelson e Cox (2008).

O que são proteínas?

Figura 1 – Mioglobina: primeira proteína a ter a sua estrutura determinada (PDB ID: 1MBN).



Fonte: RCSB Protein Data Bank (2015b).

Proteínas são cadeias formadas por polipeptídeos, que são polímeros lineares constituídos de resíduos de aminoácidos. Esses polímeros, ou biopolímeros, são macromoléculas caracterizadas pela repetição de unidades menores em sua formação, chamadas de monômeros. A Figura 1 mostra a Mioglobina, a primeira proteína a ter a sua estrutura resolvida utilizando-se a técnica de difração de raios X (KENDREW et al., 1958). Em geral, é muito comum que os termos *proteínas* e *polipeptídeos* sejam usados indistintamente, assim como *aminoácidos* e *resíduos de aminoácidos*. Contudo, moléculas referidas como polipeptídeos possuem uma massa molecular abaixo de 10.000 u, enquanto que as proteínas possuem massas moleculares maiores.

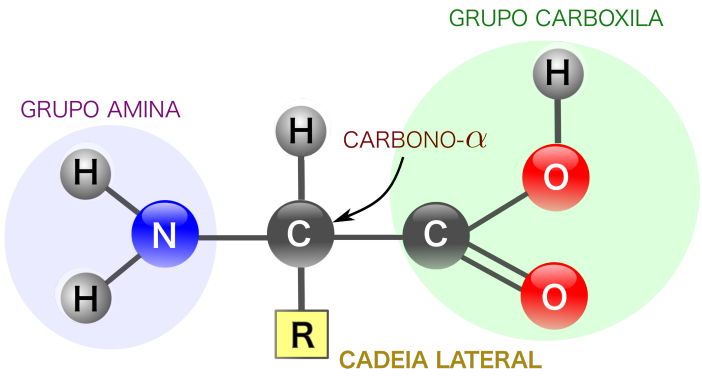
As proteínas são extremamente importantes para os seres vivos e a sua forma estrutural diz muito a respeito de suas funções bioquímicas no organismo. Entretanto, não se conhece ainda como ocorre o processo exato de enovelamento. Talvez o aspecto mais

fundamental consiste em responder à seguinte questão: conhecendo-se somente a sequência de aminoácidos de uma proteína, é possível predizer a sua estrutura tridimensional?

2.1 Aminoácidos

Aminoácidos são moléculas orgânicas que contém um grupo amina, um grupo carboxila e uma cadeia lateral R (específica para cada aminoácido) ligadas a um mesmo carbono, denominado carbono-alfa, ou carbono-α, ou C_α (veja a Figura 2).

Figura 2 – Estrutura típica de um aminoácido.



Fonte: autor.

A composição geral dos aminoácidos é feita de carbono, oxigênio, hidrogênio e nitrogênio. Os aminoácidos se ligam através de ligações covalentes chamadas de *ligações peptídicas*, liberando uma molécula de água durante esta reação, chamada de *reação de condensação*, que é uma classe comum de reações em células vivas.

Por esta razão, é utilizado o termo *resíduos de aminoácidos* para fazer referência aos aminoácidos que perderam esta molécula de água. Não obstante, frequentemente os termos *aminoácidos* e *resíduos de aminoácidos* são usados indistintamente. Em geral, os aminoácidos que formam proteínas são denominados de alfa-aminoácidos (ou $\alpha-$aminoácidos) e possuem a seguinte fórmula geral:

$$R - CH(NH_2 - COOH)\,. \tag{1}$$

O primeiro aminoácido descoberto foi a Asparagina em 1806 e, somente depois em 1938, foi descoberto o último, a Treonina. Os aminoácidos diferem uns dos outros pela sua cadeia lateral R, que pode variar em estrutura, tamanho e carga elétrica. Existem no total 20 aminoácidos naturais formadores de proteínas, no entanto, cerca de 300 outros aminoácidos já foram encontrados em células. Muitos desses aminoácidos são criados pela

modificação dos resíduos já incorporados no polipeptídeo e exercem uma variedade de funções, mas nem todos são constituintes de proteínas.

A seguir, a Tabela 1 lista os 20 aminoácidos naturais conhecidos com os seus respectivos códigos de identificação. A Tabela 2 mostra a nomenclatura e estrutura química linear de cada um deles. Convém salientar que o código de uma letra foi idealizado por Margaret Oakley Dayhoff (1925 – 1983), considerada por muitos como a fundadora do campo da *Bioinformática*.

Tabela 1 – Lista dos 20 aminoácidos naturais.

Aminoácidos	**Código de 3 letras**	**Código de 1 letra**	**Peso Molecular (g/mol)**
Alanina	Ala	A	89.0935
Cisteína	Cys ou Cis	C	121.1590
Ácido Aspártico ou Aspartato	Asp	D	133.1032
Ácido Glutâmico ou Glutamato	Glu	E	147.1299
Fenilalanina	Phe ou Fe	F	165.1900
Glicina ou Glicocola	Gly, Gli	G	75.0669
Histidina	His	H	155.1552
Isoleucina	Ile	I	131.1736
Lisina	Lys ou Lis	K	146.1882
Leucina	Leu	L	131.1736
Metionina	Met	M	149.2124
Asparagina	Asn	N	132.1184
Prolina	Pro	P	115.1310
Glutamina	Gln	Q	146.1451
Arginina	Arg	R	174.2017
Serina	Ser	S	105.0930
Treonina	Thr ou The	T	119.1197
Valina	Val	V	117.1469
Triptofano	Trp ou Tri	W	204.2262
Tirosina	Tyr ou Tir	Y	181.1894

Fonte: autor.

Tabela 2 – Nomenclatura e fórmula estrutural linear dos 20 aminoácidos naturais.

Aminoácidos	**Nomenclatura IUPAC**	**Estrutura linear / Fórmula química**
Alanina	2-aminopropiônico ou 2-amino-propanóico	CH3-CH(NH2)-COOH / $C_3H_7NO_2$
Cisteína	2-bis-(2-amino-propiônico)-3-dissulfeto ou 3-tiol-2-amino-propanóico	HS-CH2-CH(NH2)-COOH / $C_3H_7NO_2S$
Ácido Aspártico ou Aspartato	2-aminossuccínico ou 2-amino-butanodióico	HOOC-CH2-CH(NH2)-COOH / $C_4H_7NO_4$
Ácido Glutâmico ou Glutamato	2-aminoglutárico	HOOC-(CH2)2-CH(NH2)-COOH / $C_5H_9NO_4$
Fenilalanina	2-amino-3-fenil-propiônico ou 2-amino-3-fenil-propanóico	Ph-CH2-CH(NH2)-COOH / $C_9H_{11}NO_2$
Glicina	2-aminoacético	NH2-CH2-COOH / $C_2H_5NO_2$
Histidina	2-amino-3-imidazolpropiônico	NH-CH=N-CH=C-CH2-CH(NH2)-COOH / $C_6H_9N_3O_2$
Isoleucina	2-amino-3-metil-n-valérico ou 2-amino-3-metil-pentanóico	CH3-CH2-CH(CH3)-CH(NH2)-COOH / $C_6H_{13}NO_2$
Lisina	2, 6-diaminoexanóico	H2N-(CH2)4-CH(NH2)-COOH / $C_6H_{14}N_2O_2$
Leucina	2-amino-4-metil-pentanóico	(CH3)2-CH-CH2-CH(NH2)-COOH / $C_6H_{13}NO_2$
Metionina	2-amino-3-metiltio-n-butírico	CH3-S-(CH2)2-CH(NH2)-COOH / $C_5H_{11}NO_2S$
Asparagina	2-aminossuccionâmico	H2N-CO-CH2-CH(NH2)-COOH / $C_4H_8N_2O_3$
Prolina	pirrolidino-2-carboxílico	NH-(CH2)3-CH-COOH / $C_5H_9NO_2$
Glutamina	2-aminoglutarâmico	H2N-CO-(CH2)2-CH(NH2)-COOH / $C_5H_{10}N_2O_3$
Arginina	2-amino-4-guanidina-n-valérico	HN=C(NH2)-NH-(CH2)3-CH(NH2)-COOH / $C_6H_{14}N_4O_2$

Continua na próxima página...

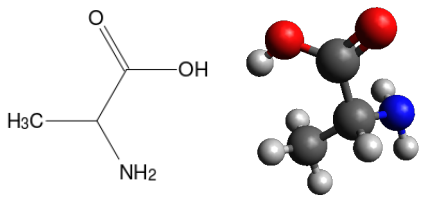
Tabela 2 – continuação da página anterior.

Aminoácidos	Nomenclatura IUPAC	Estrutura linear / Fórmula química
Serina	2-amino-3-hidroxi-propanóico	HO-CH2-CH(NH2)-COOH / $C_3H_7NO_3$
Treonina	2-amino-3-hidroxi-n-butírico	CH3-CH(OH)-CH(NH2)-COOH / $C_4H_9NO_3$
Valina	2-amino-3-metil-butanóico	(CH3)2-CH-CH(NH2)-COOH / $C_5H_{11}NO_2$
Triptofano	2-amino-3-indolpropiônico	Ph-NH-CH=C-CH2-CH(NH2)-COOH / $C_{11}H_{12}N_2O_2$
Tirosina	2-amino-3-(p-hidroxifenil)propiônico ou paraidroxifenilalanina	HO-p-Ph-CH2-CH(NH2)-COOH / $C_9H_{11}NO_3$

Fonte: autor.

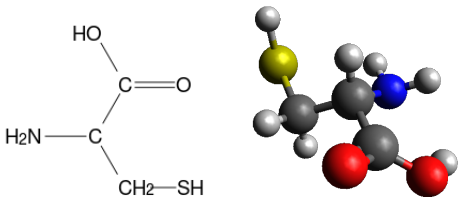
As figuras a seguir ilustram as estruturas químicas dos 20 aminácidos, tanto na representação globular quanto em bastão:

Figura 3 – Alanina.



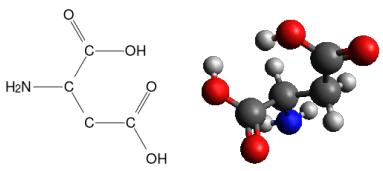
Fonte: autor.

Figura 4 – Cisteína.



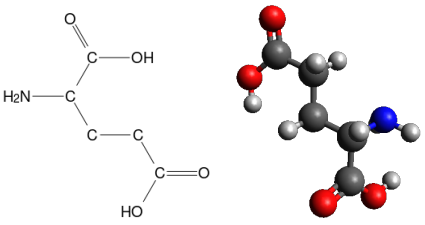
Fonte: autor.

Figura 5 – Aspartato.



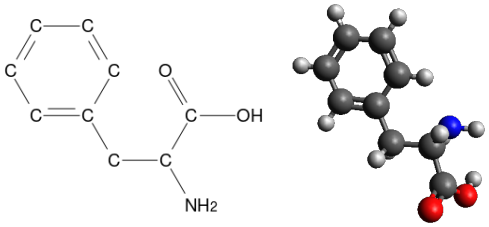
Fonte: autor.

Figura 6 – Glutamato.



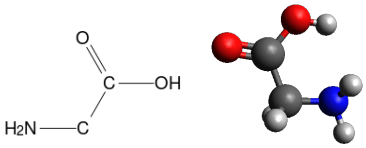
Fonte: autor.

Figura 7 – Fenilalanina.



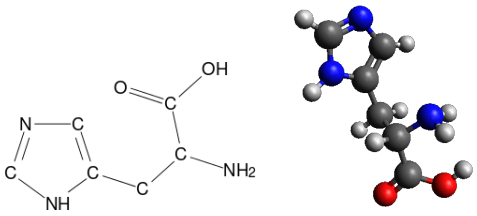
Fonte: autor.

Figura 8 – Glicina.



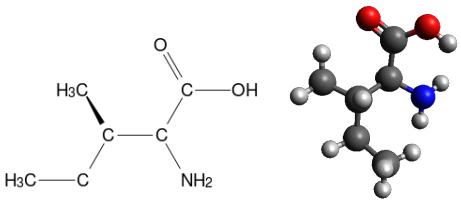
Fonte: autor.

Figura 9 – Histidina.



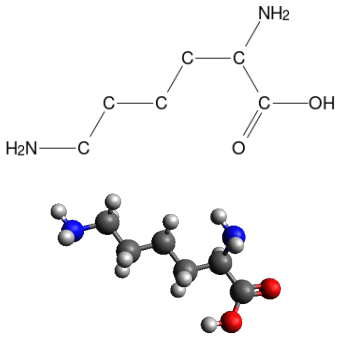
Fonte: autor.

Figura 10 – Isoleucina.



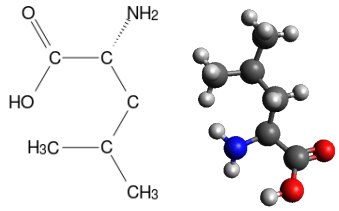
Fonte: autor.

Figura 11 – Lisina.



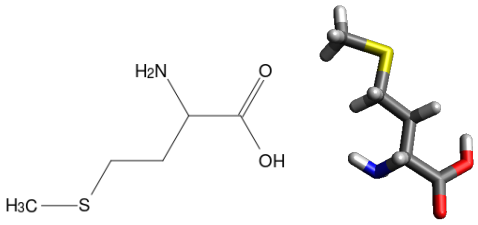
Fonte: autor.

Figura 12 – Leucina.



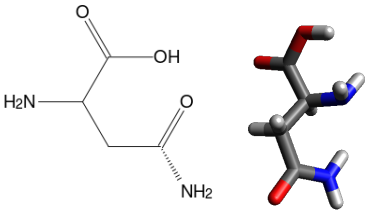
Fonte: autor.

Figura 13 – Metionina.



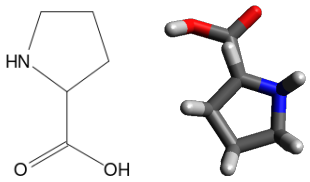
Fonte: autor.

Figura 14 – Asparagina.



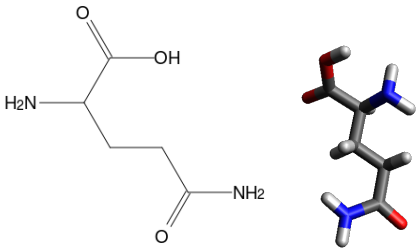
Fonte: autor.

Figura 15 – Prolina.



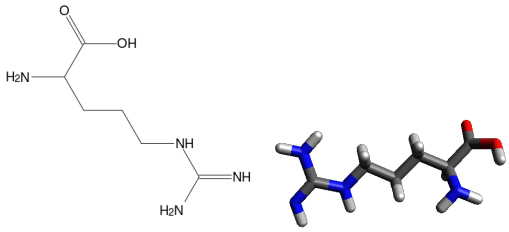
Fonte: autor.

Figura 16 – Glutamina.



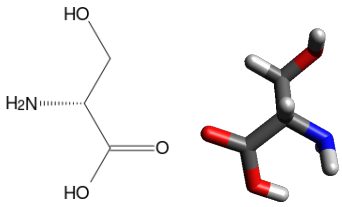
Fonte: autor.

Figura 17 – Arginina.



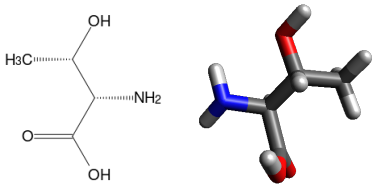
Fonte: autor.

Figura 18 – Serina.



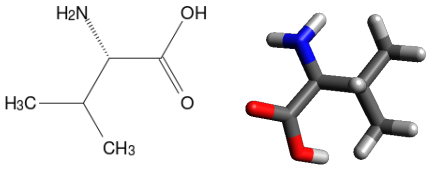
Fonte: autor.

Figura 19 – Treonina.



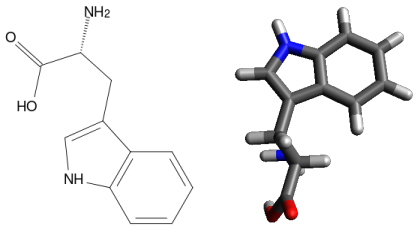
Fonte: autor.

Figura 20 – Valina.



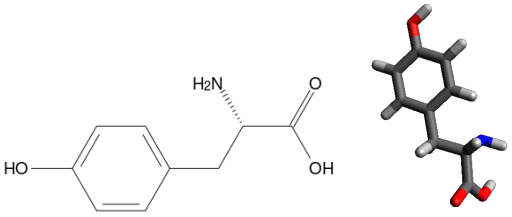
Fonte: autor.

Figura 21 – Triptofano.



Fonte: autor.

Figura 22 – Tirosina.



Fonte: autor.

2.1.1 Classificação dos Aminoácidos

Os aminoácidos podem ser classificados em três diferentes classes de acordo com as características químicas de suas cadeias laterais. A primeira classe são os aminoácidos **hidrofóbicos**, ou apolares. A segunda classe são os **hidrofílicos**, ou polares. E a terceira classe são os que apresentam, ao mesmo tempo, características polares e apolares, denominados **anfipáticos**. A Tabela 3 exibe a classificação dos aminoácidos segundo as características de suas cadeias laterais.

Tabela 3 – Classificação geral dos aminoácidos de acordo com a característica da cadeia lateral.

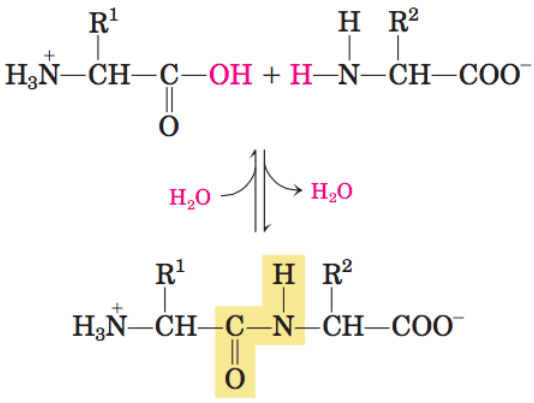
Hidrofóbicos	Hidrofílicos	Anfipáticos
val	asn	gly
leu	glu	ala
ile	gln	ser
met	his	thr
phe	lys	tyr
—	arg	trp
—	—	cys
—	—	pro

Fonte: autor.

2.1.2 Ligações Peptídicas

A ligação peptídica é um tipo de ligação covalente entre o grupo amina de um aminoácido com o grupo carboxila do outro. Quando esta ligação acontece, ocorre a liberação de uma molécula de água conforme ilustra a Figura 23. Devido a essa perda, estes aminoácidos passam a ser chamados de *resíduos de aminoácidos*. À medida que muitos aminoácidos se unem, são formadas longas cadeias denominadas polipeptídeos e estes, por sua vez, formam as proteínas. Sabe-se que a composição das proteínas é constituída por milhares de polipeptídicos.

Figura 23 – Processo de formação da ligação peptídica, com a liberação de uma molécula de água.

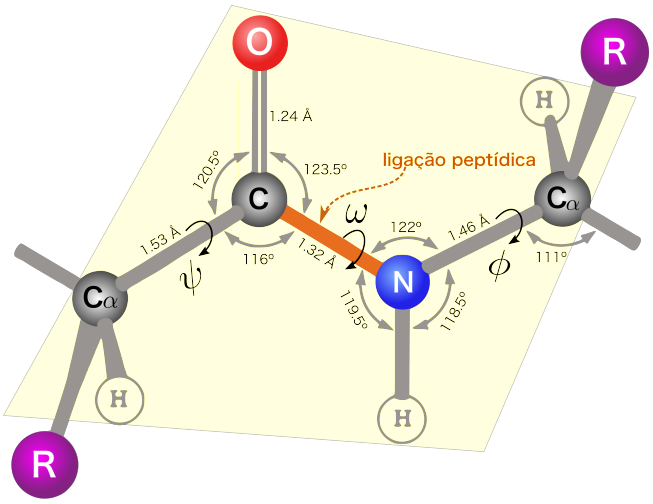


Fonte: Nelson e Cox (2008).

Na década de 1930, Linus Pauling e Robert Corey iniciaram uma série de estudos sobre a geometria e dimensão das ligações peptídicas em estruturas cristalinas de moléculas. Utilizando técnicas de difração de raios X, Pauling e Corey concluíram que o comprimento das ligações peptídicas C — N é menor do que as ligações C — N em simples compostos aminas, o que indica uma possível ressonância ou compartilhamento parcial de dois pares de elétrons entre o carbono do grupo carboxila e o nitrogênio do grupo amina. Deste modo, as ligações peptídicas C — N apresentam um caráter de ligação dupla, são rígidas e não podem rotacionar livremente, ao passo que não existem restrições para rotações entre os pares $N-C_\alpha$ e $C_\alpha-C$, representadas pelos ângulos diedros[1] ϕ e ψ, respectivamente. Assim, a cadeia principal da ligação peptídica, também conhecida como *backbone*[2], está situada em uma série de planos rígidos onde todos os C_α adjacentes são coplanares.

A Figura 24 mostra os três ângulos diedros ϕ, ψ e ω do *backbone* (também conhecidos como ângulos torcionais[3]), além dos comprimentos das ligações e dos ângulos entre os átomos, onde são indicados os valores considerados padrões pela comunidade científica. O plano em amarelo ilustra a área que compreende todo o *backbone*. Os ângulos diedros formados com a cadeia lateral são chamados de χ (lê-se "chi") e variam de χ_1 a χ_5 de acordo com cada ligação sucessiva ao longo da cadeia lateral.

Figura 24 – Características de uma típica ligação peptídica, com os valores considerados consenso para os ângulos e comprimentos de ligação, além dos ângulos diedros ψ, ϕ e ω.



Fonte: autor.

[1] Ângulo diedro ou diétrico, ou apenas diedro, é o ângulo formado pela intersecção de dois semiplanos com origem em uma mesma reta. Esta reta é chamada de aresta do diedro e os dois semiplanos são chamados de faces do diedro.

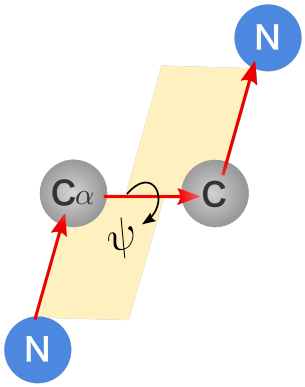
[2] Cadeia principal, ou *backbone*, compreende toda a cadeia de átomos da ligação peptídica excetuando-se as cadeias laterais.

[3] A palavra *torcional*, ou *torcionais*, não existe em nosso vocabulário, trata-se de um neologismo frequentemente empregado para referir-se a alguma coisa que sofre *torção*, neste caso, ângulos de torção.

É possível notar pela Figura 24 que o *backbone* de uma cadeia polipeptídica pode ser visto como uma série de planos rígidos, sendo que cada plano consecutivo compartilha um ponto em comum de rotação no C_α. Outra importante observação é que as sequências de C_α em uma cadeia polipeptídica são separadas por três ligações: $C_\alpha - C$, $C - N$ e $N - C_\alpha$, que se repetem ao longo de toda a cadeia principal.

Devido à rigidez das ligações peptídicas, as proteínas só podem assumir determinadas conformações espaciais que são definidas pelos ângulos diedros ϕ e ψ. Já o ângulo ω é responsável pela rotação entre o par $C - N$, podendo assumir apenas dois valores: 0° ou 180°. O sentido da rotação para os ângulos diedros é dado pela regra da mão direita; para verificar isto, veja o exemplo da Figura 25 para o ângulo ψ, que ilustra a formação das faces do diedro através de vetores representando as ligações químicas entre os átomos. Dois vetores sucessivos descrevem um plano; três vetores sucessivos descrevem dois planos; e o ângulo entre esses dois planos é o que é medido para descrever a conformação da proteína. O mesmo raciocínio se aplica para os demais diedros.

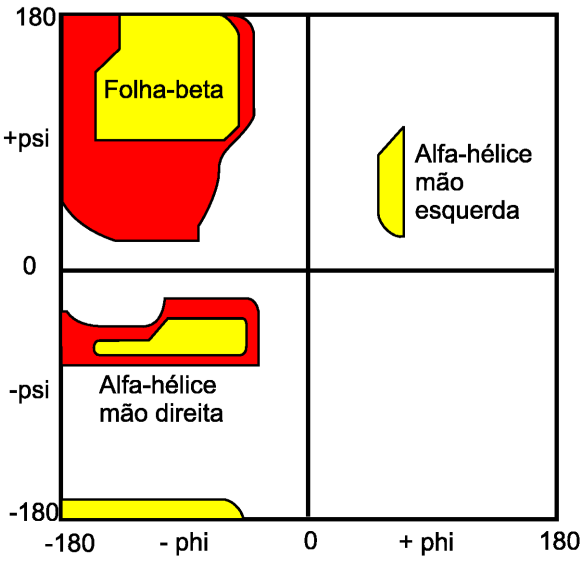
Figura 25 – Representação do diedro ψ, imaginando as ligações químicas como vetores formando dois planos (em amarelo).



Fonte: autor.

Ramachandran e Sasiskharan (1968) observaram que os ângulos ϕ e ψ concentram-se em regiões de valores específicos (Figura 26), embora, em princípio, estes ângulos poderiam assumir quaisquer valores entre $-180°$ e 180°. Todavia, vários valores são proibidos devido à interferência estérica entre os átomos da cadeia principal e os da cadeia lateral. A interferência estérica consiste da sobreposição das nuvens eletrônicas de átomos quando estão muito próximos entre si, podendo afetar a forma estrutural de uma molécula. Contudo, para alguns aminoácidos como a Glicina, em razão de sua cadeia lateral ser muito simples (apenas um átomo de hidrogênio), este mapa já não se aplica. Neste caso, a interferência estérica é bem menor e possibilita mais liberdade para a proteína assumir outros tipos de conformações.

Figura 26 – Mapa de Ramachandran.



Fonte: adaptada de Faccioli (2012).

A Figura 26 mostra um exemplo típico de um mapa de Ramachandran. Neste caso, as áreas em vermelho correspondem a conformações onde não há interferências estéricas, ou seja, são regiões em que são permitidas a ocorrência de estruturas do tipo hélices-α e folhas-β. As áreas em amarelo mostram regiões permitidas para os casos em que os átomos estão bem próximos, o que possibilita estruturas de hélices-α com orientação da mão esquerda (ver Figura 28b). Por fim, as áreas em branco são regiões em que a aproximação dos átomos é tão alta, sendo menor que a soma dos seus respectivos raios de van der Waals, que a interferência estérica proíbe qualquer tipo de conformação, exceto para a Glicina que pode ocupar todos os quadrantes do mapa (RAMACHANDRAN PLOT, 2015).

O mapa de Ramachandran, também conhecido como $[\phi, \psi]\, plot$, é muito útil para auxiliar na predição e validação de estruturas secundárias da proteína. Para cada tipo de estrutura, como as hélices-α e folhas-β, o mapa indica que existe apenas uma combinação específica para os ângulos ϕ e ψ.

2.2 Classificação Estrutural das Proteínas

No que diz respeito aos aspectos estruturais das proteínas, estas podem ser classificadas em cinco tipos:

1. **Estrutura Primária**: consiste apenas do número e da sequência de aminoácidos que constituem a proteína;

2. **Estrutura Secundária**: são "unidades" de arranjos tridimensionais, como por exemplo as hélices-α e as folhas-β, embora existam outros padrões estruturais (ou *motifs*) também importantes, a exemplo das β-turns, mas que não serão discutidas neste trabalho;

3. **Estrutura Supersecundária**: também conhecidas como *motifs* estruturais, são combinações específicas de elementos da estrutura secundária, tais como *α-helix hairpins*, *β hairpins*, $\beta - \alpha - \beta$ *motifs* e *coiled coils*;

4. **Estrutura Terciária**: é a conformação tridimensional formada pela combinação de estruturas secundárias e supersecundárias;

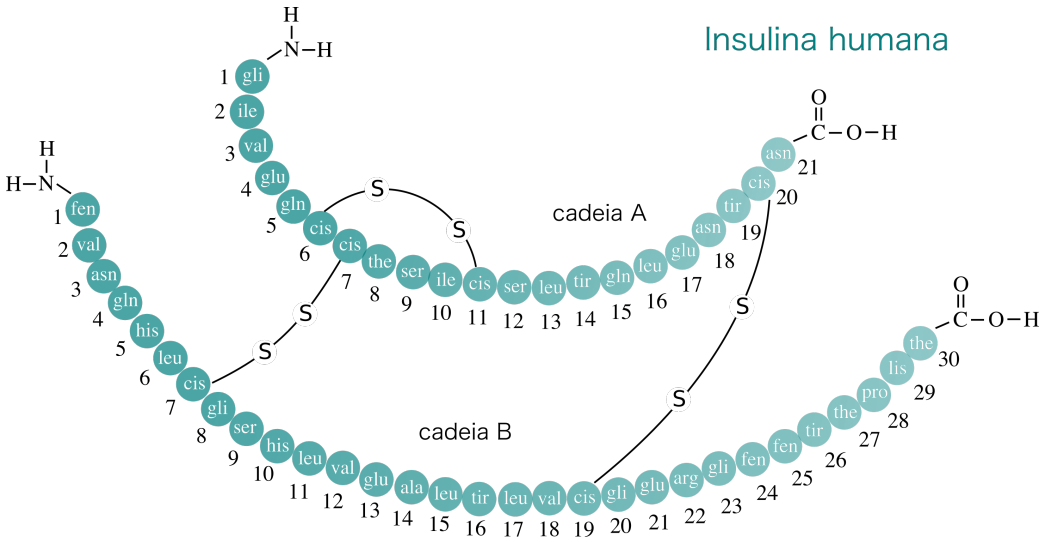
5. **Estrutura Quaternária**: refere-se ao número e à combinação de duas ou mais cadeias, ou subunidades, de proteínas, formando um complexo de multisubunidades. Exemplos: hemoglobina e o DNA polimerase.

Para este trabalho, será necessário conhecer apenas os aspectos mais relevantes de três estruturas principais: a primária, a secundária e a terciária. A seguir, uma explicação detalhada de cada uma destas estruturas.

2.2.1 Estrutura Primária de Proteínas

A estrutura primária descreve apenas o número e a sequência linear dos resíduos de aminoácidos que compõem a proteína, sem considerar nenhum aspecto de conformação espacial, consistindo apenas de uma "*string* de caracteres" de aminoácidos. A estrutura primária possui em uma extremidade um terminal amina e, na outra, um terminal carboxila. A Figura 27 mostra a representação da estrutura primária da insulina humana composta por 51 aminoácidos, com o detalhe para as indicações das pontes dissulfetos (ver seção 2.4) que contribuem para a estabilização da proteína.

Figura 27 – Estrutura primária da insulina humana composta por 51 aminoácidos.



Fonte: autor.

2.2.2 Estrutura Secundária de Proteínas

A estrutura secundária refere-se à escolha de qualquer segmento de um polipeptídeo e descreve o *arranjo espacial local* da cadeia principal, sem considerar as conformações das cadeias laterais ou a sua relação com outros segmentos. Uma característica importante das estruturas secundárias é que os diedros ϕ e ψ da cadeia principal repetem-se em padrões regulares, ou aproximadamente regular, ao longo de todo o segmento considerado.

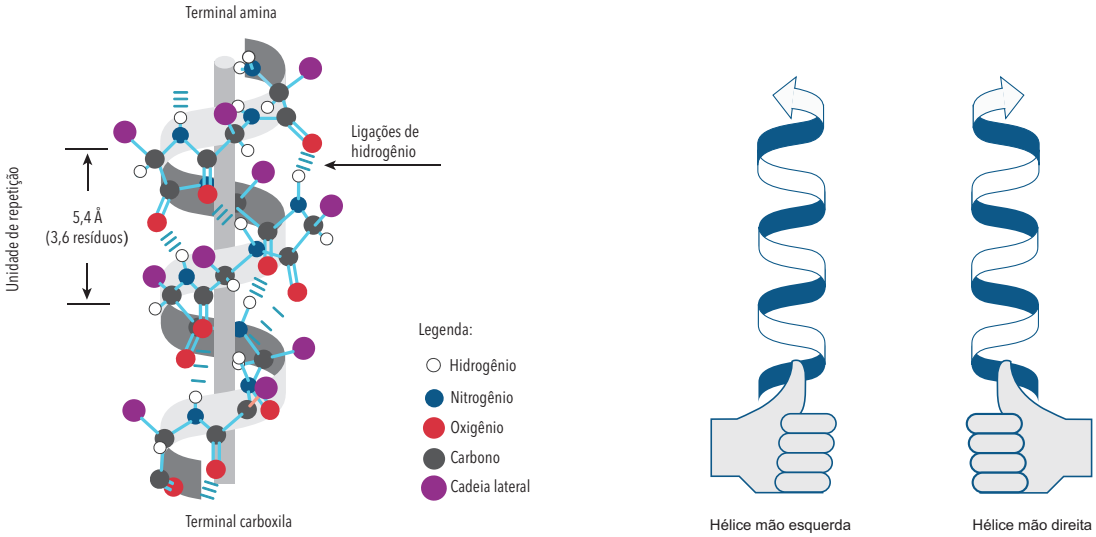
Existem dois tipos de estruturas secundárias muito comuns e estáveis que ocorrem em quase todas as proteínas: as *hélices-*α e as *folhas-*β.

Hélice-α

Consiste do arranjo mais simples que a cadeia polipeptídica pode assumir, em que os ângulos diedros são por volta de $(\phi, \psi) = (-60°, -45°)$. Nesse tipo de estrutura (Figura 28a), a cadeia principal está presa longitudinalmente a um eixo imaginário

com as cadeias laterais apontadas radialmente para fora da hélice. Em todas as proteínas, a orientação da volta da hélice é no sentido da mão direita (Figura 28b). As conformações do tipo hélice-α predominam em estruturas globulares de proteínas, englobando de 32% a 38% de todos os resíduos (CREIGHTON, 1993; KABSCH; SANDER, 1983).

Figura 28 – Estrutura secundária no formato hélice-α.



(a) Modelo de bola-bastão com as indicações das ligações de hidrogênio.

(b) Sentidos possíveis da orientação da volta da hélice: mão esquerda e mão direita.

Fonte: autor.

Folha-β

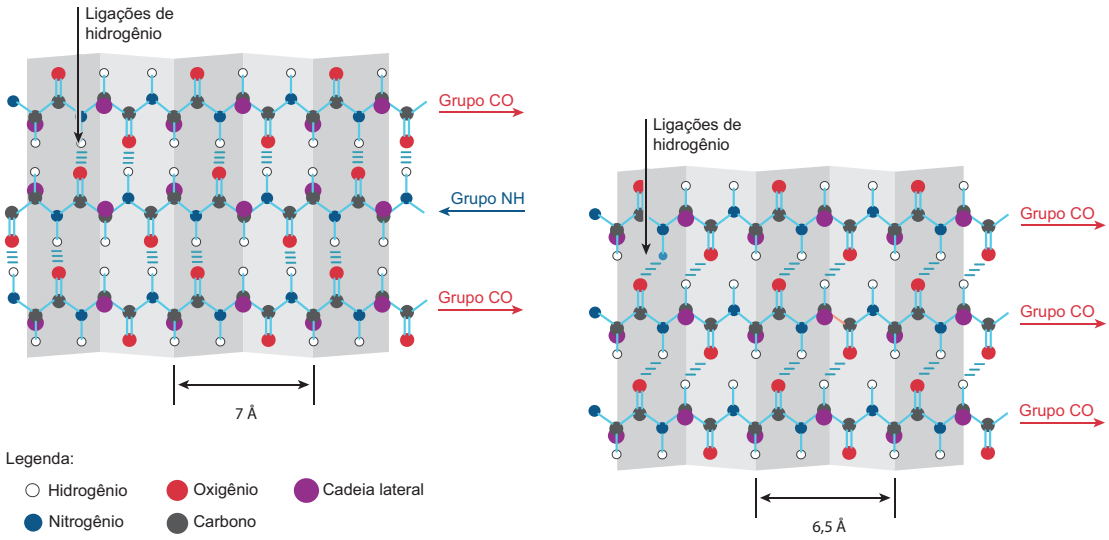
Neste tipo de conformação, a cadeia principal fica distendida em *zigzag* e os grupos do *backbone* ficam arranjados lado a lado, assumindo dois tipos de orientações:

(i) *antiparalela*: as extremidades das folhas contém grupos adjacentes distintos (Figura 29a), onde grupos CO são seguidos por grupos NH, estabelecendo entre si ligações de hidrogênio. Neste caso, os ângulos diedros são $(\phi, \psi) = (-140°, 135°)$;

(ii) *paralela*: os grupos adjacentes no final da folha são iguais (Figura 29b), entretanto, dois átomos adjacentes não formam ligações de hidrogênio entre si. Neste tipo de orientação, os ângulos diedros são tipicamente $(\phi, \psi) = (-120°, 115°)$.

Ambas as estruturas são muito similares e também possuem unidades de repetição, porém o período de repetição da orientação em paralelo (6.5 Å) é menor do que a da antiparalelo (7Å).

Figura 29 – Estrutura secundária no formato de folha-β.

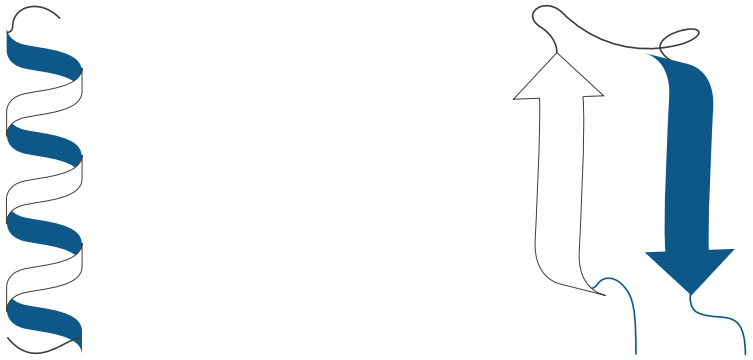


(a) Orientação antiparalela. (b) Orientação paralela.

Fonte: autor.

As folhas-β antiparalelas, por apresentarem uma orientação favorável para a proximidade das ligações de hidrogênio entre os grupos, deveria ser mais estável do que as paralelas. Entretanto, Baker e Hubbard (1984) fizeram várias pesquisas com ligações de hidrogênio e não acharam nenhuma diferença significativa na linearidade das ligações nas folhas paralelas e antiparalelas. A Figura 30 mostra as representações adotadas para as estruturas secundárias nas conformações de hélice-α e folha-β, sendo que o sentido das setas na Figura 30b é do terminal N (amina) para o terminal C (carboxila).

Figura 30 – Representação das conformações hélice-α e folha-β.



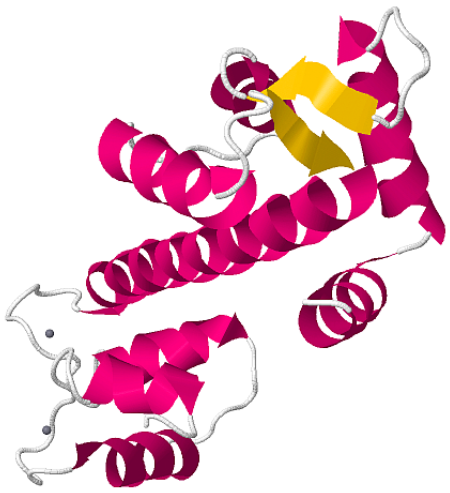
(a) Representação das hélices-α. (b) Representação das folhas-β.

Fonte: autor.

2.2.3 Estrutura Terciária de Proteínas

A estrutura terciária consiste em como os segmentos das estruturas secundárias se associam dentro de uma única cadeia polipeptídica para formar toda a estrutura tridimensional da proteína. A estrutura é estabilizada principalmente pelos efeitos hidrofóbicos, ligações de hidrogênio entre cadeias polares e forças de van der Waals. A conformação tridimensional que a proteína assume no estado de mínima energia é conhecida como **estrutura nativa**. A Figura 31 mostra a estrutura terciária da proteína PDB ID: 4TNC:

Figura 31 – Estrutura terciária da proteína PDB ID: 4TNC.



Fonte: RCSB Protein Data Bank (2015c).

Algumas proteínas podem conter duas ou mais cadeias polipeptídicas (ou subunidades), que podem ser idênticas ou diferentes. O complexo tridimensional formado por essas subunidades é chamado de *estrutura quartenária*. Assim, em consideração a essas estruturas pode-se classificar as proteínas em dois grupos maiores: *proteínas fibrosas* e *proteínas globulares*.

Proteínas fibrosas

As cadeias polipeptídicas estão em arranjos em forma de folhas ou fitas. São formadas, geralmente, de um único tipo de estrutura secundária e a sua estrutura terciária é relativamente simples. Devido à sua estrutura fibrosa, possuem a função de dar sustentação, forma e proteção externa a vertebrados.

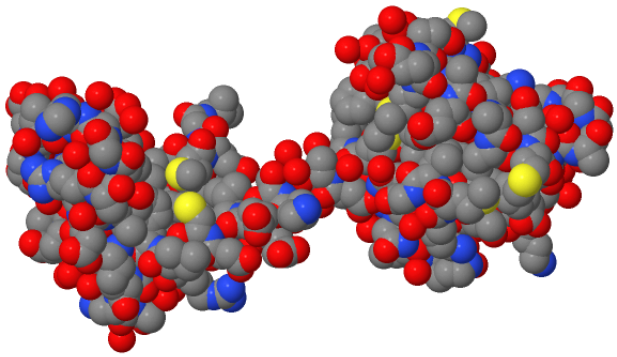
Proteínas globulares

As cadeias polipeptídicas possuem forma globular ou esférica. Normalmente contém vários tipos de estruturas secundárias em sua formação e são importantes na formação de várias enzimas e proteínas regulatórias.

Outro conceito importante em estrutura de proteínas é o de *domínio*. Introduzido por Richardson (1981), um domínio é definido como sendo a parte de uma cadeia polipeptídica que é estável e que pode se mover independentemente do restante da proteína, como se fosse uma única entidade.

Polipeptídeos com um pouco mais de algumas centenas de resíduos de aminoácidos frequentemente dobram-se em dois ou mais domínios, podendo até mesmo desempenhar funções biológicas distintas, como a de se ligar a pequenas moléculas ou mesmo a de interagir com outras proteínas. No que concerne ao tamanho, um domínio pode variar de 25 a 500 aminoácidos (LODISH et al., 2004). A Figura 32 exibe a mesma proteína da Figura 31, agora com a visualização atomística dos raios de van der Waals que evidencia a existência de dois domínios.

Figura 32 – Domínios (à esquerda e direita) da proteína PDB ID: 4TNC.



Fonte: RCSB Protein Data Bank (2015c).

2.3 O Mecanismo de *Folding* de Proteínas

O ***folding*** de proteínas, ou **enovelamento**, é um processo pelo qual a proteína passa por dobramentos sucessivos sobre si mesma, assumindo uma estrutura tridimensional característica que resulta em sua configuração biologicamente ativa, chamada de **estrutura nativa**. O processo inverso chama-se **desnaturação**, a proteína retrocede para a sua estrutura primária de aminoácidos, tornando-se uma cadeia amorfa, podendo ainda conservar pequenas estruturas enoveladas, mas sem função biológica ativa.

Experimentos mostram que *a desnaturação é um processo reversível*. Certas proteínas globulares desnaturadas pelo calor, altos pH ou reagentes de desnaturação, conseguem novamente enovelar-se para a sua estrutura nativa e voltam a realizar as suas atividades biológicas, no que é conhecido como **renaturação**. Portanto, devido à reversibilidade da desnaturação, admite-se que a estrutura terciária de proteínas pode ser completamente determinada apenas conhecendo-se a sequência de seus aminoácidos

constituintes, uma vez mantidas as condições ambientais de estabilidade na qual o *folding* ocorre. Esta premissa é conhecida como o *Dogma de Anfinsen*, também chamada de *A Hipótese Termodinâmica*, que acabou tornando-se um postulado em biologia molecular válido, pelo menos, para pequenas proteínas globulares.

Christian B. Anfinsen ganhou o prêmio Nobel de Química em 1972 pelo "seu trabalho sobre a ribonuclease, especificamente no que concerne à conexão entre a sequência de aminoácidos e a confirmação de sua atividade biológica." (NOBELPRIZE.ORG, 2015). Anfinsen (1973) demonstrou experimentalmente a desnaturação e renaturação da ribonuclease A. *Esta foi a primeira evidência de que a sequência de aminoácidos de uma cadeia polipeptídica contém toda a informação necessária para o processo de folding*, o que resulta na formação do arranjo tridimensional da proteína e, por conseguinte, em sua estrutura nativa.

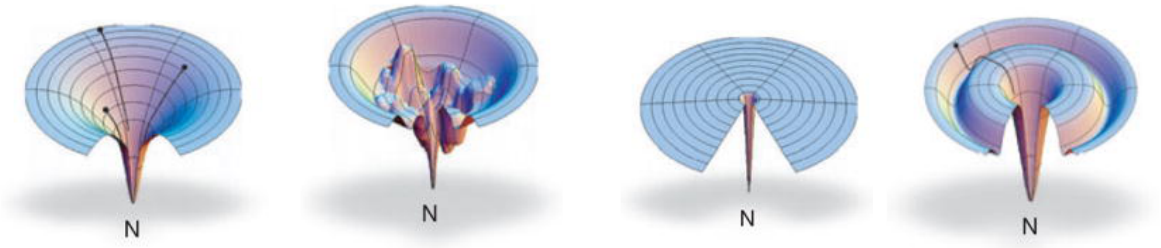
Entretanto, o processo exato do *folding* de proteínas ainda não é completamente conhecido. Levinthal (1968) levantou a seguinte situação: imagine, por exemplo, que as células como as da bactéria *E. Coli* sejam capazes de formar proteínas biologicamente ativas com 100 resíduos de aminoácidos em, aproximadamente, 5 segundos a $37°C$. Considerando, hipoteticamente, que cada aminoácido possa assumir, em média, 10 tipos de conformações diferentes, então 100 aminoácidos poderão formar até 10^{100} conformações polipeptídicas distintas. Considere ainda que o *folding* seja espontâneo devido a um processo aleatório no qual ele exauri, por tentativa e erro, todas as conformações possíveis até encontrar a mais estável.

Se cada tentativa de conformação fosse executada em um tempo biológico curto de $\approx 10^{-3}s$, então levaria cerca de 10^{77} anos para passar por todas as conformações possíveis. Estima-se que o universo, desde o início do *Big Bang*, tenha cerca de 13,7 bilhões de anos, ou $\approx 10^{10}$ anos. Portanto, fica evidente que o *folding* de proteínas não é um processo completamente randômico, baseado em tentativa e erro, que demora mais do que a idade do universo para encontrar a sua estrutura nativa. Este problema ficou conhecido como *O Paradoxo de Levinthal*. Contudo, tal paradoxo tem sido questionado uma vez que a escala de tempo pode ser significativamente reduzida (ZWANZIG; SZABO; BAGCHI, 1992).

Levinthal (1968) argumenta que deve existir uma série de passos, ou *caminhos de fold*, que o mecanismo de enovelamento deve percorrer, guiando as mudanças conformacionais até chegar em sua estrutura nativa. Existem vários modelos plausíveis para explicar o mecanismo de *folding*. Sob o ponto de vista termodinâmico, o processo de *folding* é visto como "uma trajetória afunilada na superfície de energia livre. Nesta visão, os estados desnovelados apresentam uma alta energia livre e, por outro lado, o estado nativo apresenta uma baixa energia livre." (FACCIOLI, 2012, p. 18). Pande e Rokhsar (1999) demonstraram, por meio de simulações computacionais, que uma proteína percorre vários caminhos intermediários até encontrar a sua estrutura nativa. Dill et al. (2008) argumenta que "o *folding* é uma transição da desordem para a ordem, não de uma estrutura para outra".

A Figura 33 mostra representações de perfis energéticos durante o processo de enovelamento, exibindo informações como os tipos de caminhos de *folding*, a velocidade com que são percorridos, a presença de armadilhas cinéticas, superfícies equipotenciais, entre outros. A Figura 33a ilustra um processo de *folding* rápido em que não há barreiras energéticas entre as conformações não-nativas e a nativa. A Figura 33b apresenta caminhos com armadilhas cinéticas, com possíveis caminhos intermediários fora do caminho de *folding*. Na Figura 33c, o *folding* ocorre muito lentamente, a proteína dispende bastante tempo procurando pelas conformações mais estáveis, uma vez que vários caminhos possuem a mesma energia. No caso do perfil energético da Figura 33d, existirá sempre um caminho intermediário obrigatório.

Figura 33 – Perfil energético do mecanismo de *folding*, em que N representa o ponto da estrutura nativa.



(a) *Folding* rápido. (b) Armadilhas cinéticas. (c) *Folding* lento. (d) *Folding* com um intermediário obrigatório.

Fonte: adaptada de Dill et al. (2008).

Segmentos adjacentes na sequência primária de aminoácidos tendem a continuar adjacentes nas estruturas enoveladas, embora segmentos distantes na cadeia polipeptídica podem torna-se próximos na estrutura terciária. Convém ainda ressaltar que nem todas as proteína enovelam de maneira espontânea quando são sintetizadas nas células. O mecanismo de *folding* de muitas proteínas dependem de outras proteínas denominadas **chaperonas**, que interagem com polipeptídeos parcialmente enovelados ou enovelados incorretamente, contribuindo para o correto caminho de *folding*, ou provendo pequenas condições ambientais que favorecem a ocorrência do *fold*.

O mecanismo de *folding*, também conhecido como **colapso hidrofóbico**, envolve tanto aspectos energéticos quanto estruturais. Em geral, os aspectos mais relevantes são: a energia potencial da proteína, a área apolar de acessibilidade ao solvente (aSASA), a área polar de acessibilidade ao solvente (pSASA), a área total de acessibilidade ao solvente (tSASA), o número de ligações de hidrogênio intra-proteína (HB), o raio de giro da proteína (RG), energia de solvatação (SOL) e restrições de volume.

Na estrutura primária, os aminoácidos não apresentam nenhuma conformação organizada e possuem vários graus de liberdade, com certas restrições impostas apenas

pelas ligações peptídicas entre eles. Quando o processo de *folding* se inicia, aumenta o número de contatos e de interações da proteína com ela mesma em razão dos sucessivos dobramentos e da proximidade entre os resíduos. A proteína então percorre um caminho de *folding* até encontrar a sua estrutura nativa. Durante a transição da estrutura primária para a nativa, a **entropia conformacional** diminui, uma vez que o número de graus de liberdade decresce em favor da estabilidade estrutural (DILL; BROMBERG, 2002).

A capacidade da proteína de realizar o máximo de contato possível com ela mesma denomina-se **empacotamento**, segundo Faccioli (2012):

> [...] uma proteína, a qual contém um grau máximo de empacotamento, possui todos os resíduos e têm tantos vizinhos próximos pertencentes à cadeia peptídica quanto possível, resultando-se em perfeito encaixe das cadeias laterais. (FACCIOLI, 2012, p. 25).

O empacotamento também contribui para a estabilidade estrutural da proteína, pois a estrutura nativa encontra-se no seu máximo estado estável de empacotamento. Seeliger e Groot (2007) demonstraram que todas as proteínas apresentam um alto grau de empacotamento, não importando o seu tamanho, estrutura ou função.

2.4 As Forças Indutoras do Mecanismo de *Folding*

A estabilidade da proteína depende das principais forças indutoras do processo de *folding*, todas sendo de natureza eletromagnética. As interações não-covalentes, tais como as ligações de hidrogênio, as ligações iônicas (*pontes salinas*), o efeito hidrofóbico e as interações de van der Waals (por exemplo, a força de dispersão de London), são bem mais fracas do que as ligações covalentes, entretanto, devido ao fato de que ocorrem inúmeras vezes, o efeito cumulativo garante a sua predominância para a estabilidade estrutural.

Segundo Nelson e Cox (2008, p. 114), o "termo **estabilidade** pode ser definido como a tendência de manter a conformação nativa". Contudo, as proteínas em sua estrutura nativa são fracamente estáveis, a diferença de energia livre (ΔG) separando um estado enovelado de um não-enovelado está entre 20 a 65 kJ/mol apenas. Para macromoléculas, a estrutura mais estável, ou seja, a estrutura nativa, é o estado em que a ocorrência de interações fracas é máxima. A única interação covalente que influencia significativamente no processo de *folding* são as pontes dissulfetos. A seguir, uma explicação destas principais forças:

Efeito Hidrofóbico

Durante o processo de *folding*, o efeito hidrofóbico é considerado o mais importante. O efeito consiste na tendência apresentada por substâncias apolares de se agregarem quando imersas em soluções aquosas, repelindo a água. Aminoácidos hidrofóbicos (Tabela 3), ou que apresentam uma parte hidrofóbica (anfipáticos), agrupam-se

no interior da proteína, formando um núcleo hidrofóbico no centro da proteína enovelada, enquanto a superfície externa contém a maioria dos resíduos polares. Li, Tang e Wingreen (1997) demonstraram, matematicamente, que o efeito hidrofóbico origina a principal força indutora no mecanismo de *folding*.

Ligações de Hidrogênio

A ligação de hidrogênio é um tipo de ligação eletrostática entre moléculas polares que acontece quando um grupo doador possui um átomo de hidrogênio que se liga a um átomo de alta eletronegatividade de um grupo receptor, como o flúor, nitrogênio ou oxigênio. Esse tipo de ligação pode ocorrer dentro da própria molécula (intramolecular), ou entre moléculas distintas (intermolecular). As ligações de hidrogênio favorecem a aproximação das cadeias laterais próximas aos grupos (ver Figura 28), aumentando o nível de empacotamento local. Também têm papel importante nas configurações das folhas-β (ver Figura 29).

Forças de van der Waals

As forças de van der Waals consistem em interações intermoleculares, que não sejam devidas a ligações covalentes ou interações eletrostáticas entre íons. Caracterizam-se por apresentar o seguinte comportamento: para grandes distâncias a força apresenta um caráter atrativo, enquanto que para distâncias curtas a força tem caráter repulsivo. O potencial de Lennard-Jones (detalhes no cap. 5, subseção 4.3.3), também conhecido como potencial L-J ou potencial 6-12, é frequentemente empregado para descrever este comportamento, cuja forma é:

$$U(r) = \frac{A}{r^{12}} - \frac{B}{r^{6}} \ . \tag{2}$$

As forças dividem-se em três interações distintas:

1. Força entre dois dipolos permanentes: *força de Keesom*;
2. Força entre um dipolo permanente e um dipolo induzido: *força de Debye*;
3. Força entre dois dipolos instantaneamente induzidos: *força de dispersão de London*.

As forças de van der Waals são mais fracas que as ligações de hidrogênio e interações dipolo-dipolo.

Forças de Dispersão de London

A força de dispersão de London é uma força intermolecular atrativa fraca, que induz a formação temporária de um dipolo instantâneo entre duas moléculas apolares. A força de dispersão de London é um caso particular das forças de van der Waals. Dill e Bromberg (2002) estabeleceram uma relação entre o efeito hidrofóbico e a

força de London: quanto maior o empacotamento devido às interações hidrofóbicas, mais contatos são formados entre as cadeias laterais apolares, permitindo que as interações de London possam se estabelecer.

Pontes Salinas

As pontes salinas são interações iônicas que surgem entre uma cadeia lateral carregada positivamente e outra carregada negativamente. As pontes salinas de maior contribuição à estabilidade estrutural são aquelas formadas entre grupos de íons no núcleo hidrofóbico, favorecendo a criação de um ambiente essencialmente apolar. Isto auxilia fortemente na especificidade de uma conformação, desestabilizando aquelas nas quais as interações entre os íons não sejam ótimas (DILL; BROMBERG, 2002).

Ponte Dissulfeto

A ponte dissulfeto, ou ligação dissulfeto, também conhecida como ligação S-S (entre dois átomos de enxofre), é a única ligação covalente a contribuir no processo de *folding*, favorecendo a estabilidade estrutural. Ela se origina da interação entre dois grupos tiol (-SH) das cadeias laterais de resíduos de Cisteínas. São raramente encontradas em proteínas intracelulares, sendo mais frequentes em proteínas secretadas para o meio extracelular, como por exemplo a insulina (ver Figura 27).

2.5 Principais Métodos de Determinação de Estruturas de Proteínas

Dentre os métodos de determinação e análise de estrutura de proteínas, três merecem destaque: a difração de raios X, a ressonância magnética nuclear e os métodos computacionais de predição.

2.5.1 Cristalografia de Difração de Raios X

No caso da cristalografia de difração de raios X, a proteína precisa estar cristalizada e isto nem sempre é possível, pois é preciso que uma série de condições sejam satisfeitas, tais como o pH, a temperatura, a concentração da proteína e a natureza do solvente. Tais restrições limitam o uso desta técnica, uma vez que não é fácil predizer boa condições para a cristalização da proteína (DRENTH, 1994).

Este método é utilizado para identificar a posição dos átomos da rede cristalina e seu princípio de funcionamento é bem simples: quando um feixe de raios X atinge os átomos da rede, ocorre difração e os raios são espalhados em direções muito específicas, dada pela *Lei de Bragg*. Então, medindo-se os ângulos e as intensidades dos feixes difratados, é possível produzir um mapa tridimensional da densidade de elétrons dentro

do cristal. As regiões com grande densidade eletrônica revelam as posições médias dos núcleos atômicos, o que possibilita reconstruir a estrutura final da proteína.

2.5.2 Ressonância Magnética Nuclear (RMN)

No método de RMN a proteína precisa apenas estar em solução, o que torna a técnica mais abrangente do que a difração por raios X. A ressonância magnética nuclear funciona baseada nas propriedades quânticas dos *spins* dos átomos. Aplicando-se fortes campos magnéticos externos, os *spins* formam pequenos dipolos magnéticos que se alinham na direção do campo em dois sentidos possíveis: paralelo (baixa energia) e antiparalelo (alta energia). Quando aplicado um pulso eletromagnético curto com uma determinada frequência de ressonância, a energia é absorvida e depois emitida nas transições entre os níveis de energia rotacionais dos núcleos, então o espectro de absorção resultante fornece vários tipos de informações importantes, como por exemplo a distância entre as ligações químicas.

Entretanto, a análise estrutural de proteínas foi apenas possível graças ao surgimento de técnicas de RMN bidimensionais. Para gerar as estruturas tridimensionais são necessárias informações adicionais, como a geometria, a quiralidade, o comprimento das ligações, os ângulos de ligação e o tamanho das esferas de van der Waals. Depois é feito um processamento computacional e são geradas famílias de estruturas correlacionadas, correspondendo a um intervalo de conformações possíveis. Umas das desvantagens deste método é que sua utilização restringe-se a pequenas moléculas de proteínas (BRANDEN; TOOZE, 1991).

2.5.3 Métodos Computacionais de Predição

Nem sempre é possível dispor de equipamentos para realizar experimentos de RMN e difração de raios X, além do custo alto e do tempo gasto na preparação das amostras, como no caso de se cristalizar uma proteína. Assim, alternativas a esses experimentos de laboratório fizeram surgir tentativas de predição de estruturas de proteínas por meio de métodos computacionais (*in silico*), os quais podem ser divididos em duas categorias principais: *template-based modelling* e *template free modelling*.

Template-based modelling

Neste modelo, os algoritmos podem empregar estruturas terciárias já conhecidas para realizar a predição. Este método é dependente da acurácia do alinhamento, do refinamento do modelo e da qualidade das estruturas conhecidas (GINALSKI, 2006). Entre os métodos existentes, destacam-se os de *homologia* e *threading* (HILBERT; BÖHM; JAENICKE, 1993).

Template free modelling

Este modelo não depende de nenhum conhecimento prévio das estruturas terciárias, uma vez dada a sequência alvo, são utilizados modelos físicos para derivar as informações necessárias para simular o *folding*. Dentre os modelos que utilizam princípios físicos, destaca-se o método *ab initio* (LEE; WU; ZHANG, 2015).

Conforme mencionado no capítulo 1 (seção 1.2), os métodos *in silico*, em geral, estão muito aquém dos experimentos de bancada de laboratório com relação à acurácia na determinação das estruturas. Assim, é necessário dispor de meios de avaliação destes algoritmos a fim de mensurar a eficiência e a qualidade das predições realizadas. Um dos eventos que atendem a esta finalidade é o CASP (2015) – *Comparative Assessment of Methods for Protein Structure Prediction* – um evento mundial que ocorre a cada 2 anos, composto por vários grupos de pesquisa que delineiam o *estado da arte* dos métodos computacionais de predição de proteínas.

2.6 Considerações Finais

Neste capítulo foram apresentados os conceitos biológicos necessários à compreensão do problema da predição de proteínas. Proteínas são polipeptídeos formados por longas cadeias de aminoácidos, estes por sua vez diferem entre si por suas cadeias laterais. Ao todo, 20 aminoácidos formam todas as proteínas naturais conhecidas, embora existam outros tipos de aminoácidos também. Os aminoácidos podem ser classificados, de acordo com as características químicas da sua cadeia lateral, em hidrofóbicos, hidrofílicos ou anfipáticos.

Com relação aos aspectos estruturais das proteínas, os métodos *ab initio* usualmente consideram os ângulos diedros ϕ, ψ e χ como os parâmetros principais, ditos *parâmetros livres*, responsáveis por gerar novas conformações estruturais. Ramachandran e Sasiskharan (1968) mostraram que os diedros ϕ e ψ podem assumir apenas valores bem específicos (ver Figura 26).

Por fim, foi visto que o efeito hidrofóbico é a principal força indutora do processo de *folding*, como demonstrado matematicamente por Li, Tang e Wingreen (1997). Existem diferentes métodos de se determinar a estrutura de proteínas, os de bancada de laboratório como a cristalografia de difração de raios X e RMN, e aqueles simulados por computador (*in silico*), divididos em *template-based modelling* e *template free modelling*.

CAPÍTULO 3

Predição Computacional de Estruturas de Proteínas

3.1 Representação Computacional de Proteínas

As proteínas, devido às suas estruturas complexas, necessitam de uma representação computacional robusta. É preciso prover informações tais como os comprimentos de ligação e ângulos de ligação entre os átomos, os ângulos torcionais (diedros), além de informações acerca das posições dos átomos. Utiliza-se, frequentemente, os formatos de arquivo FASTA e PDB para guardar esses tipos de dados. Essas informações dividem-se em duas formas de representação: *coordenadas cartesianas* e *coordenadas internas*.

Coordenadas cartesianas

A proteína é representada por um sistema de coordenadas cartesianas, em que são dadas as orientações espaciais (posição tridimensional) de cada átomo que a compõe.

Coordenadas internas

A proteína é representada por uma matriz, conhecida como **Matriz-Z**, que contém informações sobre cada átomo em termos do número atômico, do comprimento de ligação entre dois átomos, do ângulo de ligação com um terceiro átomo e do valor do ângulo diedral formado com um quarto átomo.

É possível converter de um sistema de coordenadas para outro, todavia, os resultados nem sempre são aqueles esperados. Os algoritmos de conversão podem variar significativamente em sua precisão numérica e, para macromoléculas como proteínas, átomos distantes ao longo da cadeia, por vezes, encontram-se muito próximos no espaço cartesiano, então erros de arredondamento podem ir acumulando e possibilitando a ocorrência de resultados inesperados.

Koslover e Wales (2007) realizaram uma comparação da eficiência dos sistemas de coordenadas na otimização da geometria das proteínas e demonstraram que existe uma dependência em relação ao tamanho da proteína. As coordenadas internas foram mais

eficientes em proteínas pequenas, enquanto que para as outras proteínas as coordenadas cartesianas foram mais eficientes.

Reyes (2011) propõe uma outra forma de representação tridimensional de proteínas, levando-se em consideração um sistema de coordenadas esféricas (ρ, ϕ e θ), principalmente para proteínas globulares ou esféricas, apresentando duas aplicações de várias outras em potencial. Basicamente, a proteína pode ser separada em duas partes, uma camada externa e uma parte central. A parte central compreende a parte da proteína abaixo de um certo valor de corte para o raio ρ, já a camada externa é a parte restante acima deste valor. Deste modo, foi possível identificar saliências e invaginações na superfície da proteína, além de ter sido verificado que a superfície externa é muito mais rica em resíduos de aminoácidos hidrofílicos, enquanto que a parte central é mais rica em resíduos hidrofóbicos, como era de se esperar.

3.1.1 FASTA, PDB e Banco de Dados

O formato FASTA consiste de um arquivo de texto que pode representar tanto uma sequência de nucleotídeos quanto de aminoácidos. Este formato tornou-se padrão no campo da Bioinformática, contém um cabeçalho com uma linha de identificação começando com o símbolo “>”, seguido por identificadores do composto biológico em questão. A linha seguinte contém a sequência de dados dos nucleotídeos ou aminoácidos chamada de *bare sequence*, representados pelo código de uma letra (BLAST, 2015). A seguir, a Figura 34 apresenta o arquivo FASTA da proteína PDB ID: 4TNC (Figura 31).

Figura 34 – Arquivo FASTA da proteína PDB ID: 4TNC.

```
>4TNC:A|PDBID|CHAIN|SEQUENCE
TSAMDQQAEARAFLSEEMIAEFKAAFDMFDADGGGDISTKELGTVMRMLGQNPTKEELDAIIEEVDEDGSGTIDFEEFLV
MMVRQMKEDAKGKSEEELADCFRIFDKNADGFIDIEELGEILRATGEHVTEEDIEDLMKDSDKNNDGRIDFDEFLKMMEG
VQ
```

Fonte: autor.

O formato PDB, referente a *Protein Data Bank*, tornou-se o arquivo padrão para representar as coordenadas de posições dos átomos. Também consiste de um arquivo de texto, porém muito maior que o FASTA, cada linha é chamada de *record*, arranjadas em diferentes formas para descrever a estrutura da proteína. A Figura 35 mostra um trecho do início do arquivo PDB da proteína PDB ID: 4TNC.

Existem vários outros formatos de arquivos disponíveis e, tão importante quanto, são também os vários bancos de dados disponíveis para consulta de compostos biológicos, como por exemplo o RCSB Protein Data Bank (2015d), que conta ainda com o wwPDB (2015), uma organização que gerencia os arquivos PDBs para garantir a disponibilidade gratuita para todos. Convém, sobretudo, mencionar o NCBI - National Center for

Biotechnology Information (2015), referência mundial para pesquisas de informações biológicas, contendo um mecanismo de busca para uma grande gama de banco de dados.

Figura 35 – Início do arquivo PDB da proteína PDB ID: 4TNC.

```
HEADER    CONTRACTILE SYSTEM PROTEIN            28-SEP-87   4TNC
TITLE     REFINED STRUCTURE OF CHICKEN SKELETAL MUSCLE TROPONIN C IN
TITLE    2 THE TWO-CALCIUM STATE AT 2-ANGSTROMS RESOLUTION
COMPND    MOL_ID: 1;
COMPND   2 MOLECULE: TROPONIN C;
COMPND   3 CHAIN: A;
COMPND   4 ENGINEERED: YES
SOURCE    MOL_ID: 1;
SOURCE   2 ORGANISM_SCIENTIFIC: GALLUS GALLUS;
SOURCE   3 ORGANISM_COMMON: CHICKEN;
SOURCE   4 ORGANISM_TAXID: 9031
KEYWDS    CONTRACTILE SYSTEM PROTEIN
EXPDTA    X-RAY DIFFRACTION
AUTHOR    M.SUNDARALINGAM
REVDAT   4   24-FEB-09 4TNC    1       VERSN
REVDAT   3   01-APR-03 4TNC    1       JRNL
REVDAT   2   19-APR-89 4TNC    1       JRNL
REVDAT   1   16-JUL-88 4TNC    0
SPRSDE     16-JUL-88 4TNC      3TNC
```

Fonte: autor.

3.1.2 *Softwares* de Renderização e Visualização

De posse dessas informações computacionais, sejam elas informadas em arquivos FASTA, PDB, ou em outros formatos, existem diversos *softwares* que renderizam e representam visualmente as proteínas, sendo que a representação mais utilizada são os *diagramas de Richardson*, ou *diagramas de fita* (do inglês, *ribbon diagram*), como visto na Figura 31. Existem também outros tipos de representações visuais, como os raios de van der Waals (Figura 32), *wireframes*, *ball and strick*, *rockets*, entre outros.

Os programas de renderização e visualização também fornecem vários outros tipos de análises. Em RCSB Protein Data Bank (2016) existe uma grande lista destes *softwares*, dentre eles destacam-se principalmente:

1. **Jmol**: Visualizador *open-source* de estruturas químicas 3D, amplamente utilizado em sites como um *applet* Java para renderização, como em RCSB Protein Data Bank (2015d). Jmol conta ainda com recursos dedicados para química, biomoléculas, cristais e outros materiais (JMOL, 2015).

2. **PyMOL**: Visualizador molecular de alta performance com suporte a animações e renderização de alta qualidade, com rotinas de cristalografia e outras atividades moleculares gráficas usuais (PYMOL, 2015);

3. **RasMol**: Ferramenta de visualização gráfica de estruturas moleculares (RASMOL, 2015);

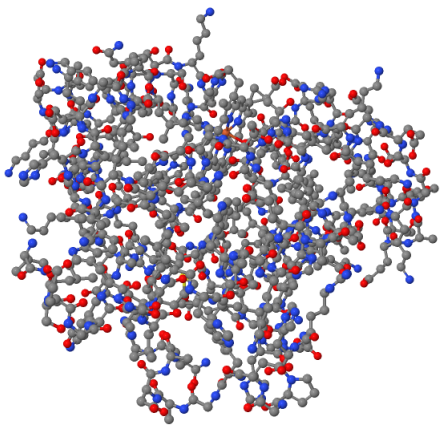
4. **VMD (Visual Molecular Dynamics)**: Programa de visualização molecular com suporte a animação 3D e análise de grandes sistemas biomoleculares (VMD, 2015);

5. **UCSF Chimera**: Programa de visualização interativa e análise de estrutura molecular (UCSF CHIMERA, 2015);

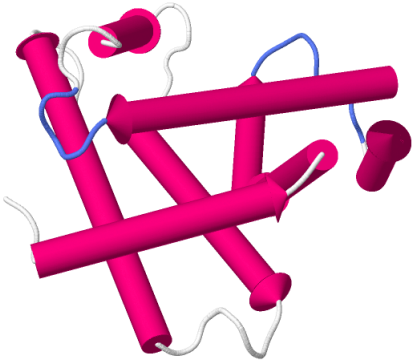
6. **Bioblender**: Programa construído com base no Blender, famoso *software open-source* de renderização 3D, sendo possível trabalhar com proteínas em 3D, visualizando a sua superfície de forma realista e determinar alguns movimentos com base na sua conformação (BIOBLENDER, 2015).

A seguir, a Figura 36 ilustra outras possibilidades de visualização estrutural da Mioglobina fornecidas pelo *software* Jmol (2015):

Figura 36 – Outros exemplos de representações estruturais da Mioglobina (PDB ID: 1MBN) renderizadas pelo *software* Jmol (2015).



(a) Representação *ball and stick*: indica as posições relativas dos átomos e das ligações químicas.



(b) Representação *rockets*: as estruturas do tipo hélices-α são representadas por cilindros com setas.

Fonte: RCSB Protein Data Bank (2015b).

3.2 Modelagem Computacional do *Folding* de Proteínas

Conforme visto no capítulo 2 (subseção 2.5.3), existem dois principais métodos de modelagem computacional para resolver o problema do PSP: *template-based modelling*, utilizando técnicas de homologia e *threading*, e *template free modelling*, cuja abordagem *ab initio*, ou primeiros princípios, utiliza os princípios físicos envolvidos no processo de *folding*. A seguir, uma explicação detalhada de cada uma destas técnicas.

3.2.1 Modelagem Comparativa ou por Homologia

A modelagem por homologia consiste em predizer a estrutura terciária de uma proteína desconhecida com base na estrutura conhecida de uma outra proteína semelhante, ou homóloga. Portanto, esta técnica é completamente dependente dos dados experimentais e não requer um alto esforço computacional (ECHENIQUE, 2007).

Uma das maneiras frequentemente empregadas de se medir a similaridade entre duas proteínas é pelo cálculo do desvio da raiz quadrática média, ou do inglês, RMSD (*Root-Mean-Square Deviation*), que consiste na medida da distância média entre os átomos de proteínas sobrepostas.

Definição 1. *Dados dois conjuntos* $\mathbf{v}$ *e* $\mathbf{w}$ *de n pontos, o RMSD é definido como:*

$$\begin{aligned} RMSD(\mathbf{v}, \mathbf{w}) &= \sqrt{\frac{1}{n}\sum_{i=1}^{n} ||\mathbf{v_i} - \mathbf{w_i}||^2} \\ &= \sqrt{\frac{1}{n}\sum_{i=1}^{n} \left[(v_{ix} - w_{ix})^2 + (v_{iy} - w_{iy})^2 + (v_{iz} - w_{iz})^2\right]}\ , \end{aligned} \tag{3}$$

em que $\mathbf{v}$ *e* $\mathbf{w}$ *são dois vetores que representam as posições dos átomos de cada sequência.*

O valor do RMSD, para sistemas biológicos, é normalmente expresso utilizando o ångström (Å) como unidade de comprimento.

Em Lessel e Schomburg (1994), a similaridade é calculada de outra maneira com base nas posições dos carbonos-α. Utilizando os dados do PDB, conseguiram dividir as proteínas em 182 famílias estruturais, sendo possível estimar quais eram as relações entre os membros de mesma classe.

Hilbert, Böhm e Jaenicke (1993) estudaram vários alinhamentos de estruturas conhecidas, com diferentes formas e classes funcionais, apresentando diferentes graus de homologia. O estudo sugeria algumas relações entre sequências homólogas e diferenças estruturais, onde alinhamentos com mais de 50% de similaridade têm 90% de seus resíduos em regiões que conservam uma mesma estrutura. Já regiões estruturalmente divergentes, porém com 50% de similaridade no alinhamento, possuem uma conformação estrutural global parecida. Verificou-se que grandes desvios estruturais podem ocorrer se a similaridade for baixa.

Kabsch e Sander (1983) demonstraram que mesmo que a similaridade seja exata para pequenos segmentos, isto ainda não fornece indicação de estrutura homóloga. Porém, com os estudos de Cohen, Presnell e Cohen (1993) sobre hexapeptídeos, foi possível demonstrar que dentro de uma classe estrutural de proteína ou domínios, a similaridade

na estrutura de um hexapeptídeo sequencialmente idêntico é preservada. Este estudo ensejou a possibilidade de desenvolver algoritmos para predizer as estruturas terciárias de proteínas com domínio conhecido (BARTON; COHEN; BRADFORD, 1993).

Kaczanowski e Zielenkiewicz (2010) destacaram que proteínas homólogas geralmente possuem estruturas terciárias semelhantes. Portanto, a eficácia da modelagem por homologia depende, sobremaneira, da qualidade dos dados experimentais acerca das estruturas conhecidas a fim de se realizar uma boa predição.

3.2.2 Modelagem por *Threading*

A modelagem por *threading*, proposta por Jones, Taylor e Thornton (1992), também é um tipo de modelagem comparativa que depende da base de dados experimentais de estruturas terciárias conhecidas. A diferença consiste quando a sequência alvo não possui, a princípio, nenhuma proteína homóloga conhecida no PDB, então o que se faz é tentar alinhar cada aminoácido da sequência alvo com *um modelo de estrutura* escolhido aleatoriamente, avaliando o quanto a sequência alvo é similar ao modelo escolhido.

Tal abordagem justifica-se devido ao número limitado de *folds* encontrados na natureza, além de que a maioria das proteínas no PDB possuem estruturas similares conhecidas. Então, dada uma sequência alvo, compara-se com um conjunto de modelos de estruturas conhecidas, assim uma função de avaliação pontua os alinhamentos levando-se em conta fatores como: a preferência pela acessibilidade ao solvente, a preferência por uma estrutura secundária em particular, interações entre segmentos vizinhos, entre outros. Por fim, um método de escolha do melhor alinhamento é empregado.

O *software* Threader (2015) tem sido utilizado por milhares de usuários desde o seu lançamento público em 1994. No primeiro CASP, o Threader (2015) foi o método de maior sucesso em predizer *folds* de proteínas, chegando a acertar 8 de 11 estruturas.

3.2.3 Modelagem *Ab Initio*

No caso da modelagem *ab initio*, também conhecida como *de novo modelling* (BRADLEY; MISURA; BAKER, 2005), *physcis-based modelling* (OŁDZIEJ et al., 2005), ou ainda, *free modelling* (JAUCH et al., 2007), a predição não depende do conhecimento prévio de nenhuma estrutura já resolvida. Portanto, não se utiliza nenhuma base de dados experimentais. Este modelo emprega leis físicas para descrever a interação da proteína com um **campo de força** e com um determinado solvente.

Este método consiste em realizar uma busca pelo espaço conformacional de acordo com uma determinada função de energia, gerando soluções candidatas. Por fim, um método de seleção adequado é responsável por escolher as estruturas que mais se aproximam do estado nativo. Como o método *ab initio* é o empregado neste trabalho, será

feita uma descrição teórica mais detalhada. A Tabela 4 mostra os principais algoritmos de modelagem *ab initio*.

Tabela 4 – Lista dos principais algoritmos de modelagem *ab initio*.

Algoritmo	**Campo de Força**	**Método de Busca**	**Modelo de Seleção**	**Tempo de CPU**
AMBER/CHARMM Brooks et al. (1983)	*Physics-based*	Dinâmica Molecular	Mínima energia	Anos
UNRES Ołdziej et al. (2005)	*Physics-based*	CSA	*Clustering*/ Energia livre	Horas
ASTRO-FOLD Klepeis e Floudas (2003)	*Physics-based*	αBB/CSA/MD	Mínima energia	Meses
ROSETTA Robbeta.org (2015)	*Physics and knowlodge-based*	Monte Carlo	*Clustering*/ Energia livre	Meses
TASSER/Chunk-TASSER CSSB Systems Biology (2015)	*Knowlodge-based*	Monte Carlo	*Clustering*/ Energia livre	Horas
I-TASSER Wu, Skolnick e Zhang (2007)	*Knowlodge-based*	Monte Carlo	*Clustering*/ Energia livre	Horas

Fonte: Lee, Wu e Zhang (2015).

Em geral, três fatores são determinantes para o sucesso deste modelo (LEE; WU; ZHANG, 2015):

1. Escolha apropriada da **função de energia**, em que a estrutura nativa da proteína corresponda ao estado termodinâmico mais estável;
2. Um **método de busca** eficiente, o qual rapidamente identifica os estados de mais baixa energia durante a busca pelo espaço conformacional;
3. **Seleção de estruturas nativas** de um conjunto de estruturas candidatas.

3.2.3.1 Funções de Energia Potencial

As funções de energia podem ser classificadas de duas formas a depender de qual tipo de abordagem é utilizada para a modelagem 3D: *physics-based* e *knowledge-based* (YANG, 2009).

Physics-based

Neste caso, a mecânica quântica deveria ser aplicada para calcular as interações entre os átomos que, por sua vez, devem ser descritos por seus tipos de átomos[4], onde apenas o número de elétrons é relevante (HAGLER; HULER; LIFSON, 1974). Contudo, simular interações quânticas exige um custo computacional muito elevado, não sendo possível para a tecnologia atual. Então, na prática, o que se faz é utilizar um campo de força contendo termos como os comprimentos de ligação, os ângulos de ligação e torcionais, interações eletrostáticas e de van der Waals, além de um grande número de tipos de átomos. Para cada um destes termos, as suas propriedades físicas e químicas devem ser suficientemente parecidas com os parâmetros da teoria da mecânica quântica ou do empacotamento de cristais.

Exemplos de campos de força bem conhecidos são: AMBER (WEINER et al., 1984), CHARMM (BROOKS et al., 1983), OPLS (JORGENSEN; TIRADO-RIVES, 1988) e GROMOS96 (GUNSTEREN et al., 1996). A principal diferença entre os campos consiste na escolha dos tipos de átomos e dos parâmetros de interação. Para o *folding* de proteínas, os campos de força são frequentemente acoplados com simulações de Dinâmica Molecular, tanto para a predição de estrutura de proteínas (PSP), quanto para o refinamento de estruturas. Entretanto, utilizar Dinâmica Molecular para o PSP não tem demonstrado muito sucesso (YANG, 2009).

Knowledge-based

Os potenciais do tipo *knowledge-based* são deduzidos de forma empírica a partir de análises estatísticas de proteínas com estruturas já resolvidas no banco de dados PDB. Segundo Skolnick (2006), um potencial deste tipo tem dois termos principais:

(i) Termos genéricos que independem da sequência alvo, como por exemplo as ligações de hidrogênio e a rigidez local do *backbone* de uma cadeia polipeptídica;

(ii) Termos que dependem dos aminoácidos ou da sequência da proteína, como por exemplo o potencial de contato entre um par de resíduos, o potencial de contato devido à interação entre os átomos e as **propensões**[5] da estrutura secundária.

[4] Tipos de átomos (*atom types*) são classificações usadas em simulações de campo de força, em que de acordo com o elemento químico e o ambiente de ligação, serve para identificar grupos funcionais, hidrogênios adicionais, determinar o raio de van der Waals e identificar as ligações de hidrogênio. Por exemplo, para o campo de força CHARMM, veja Forcefield Based Simulations (2015).

[5] Propensões em predição de estruturas de proteínas significa a *possibilidade* de que um aminoácido da sequência alvo faça parte de um certo tipo de estrutura secundária (*e.g.*, hélices-α ou folhas-β). As propensões são classificadas como altamente formadora, formadora, pouco formadora, indiferentemente formadora, não formadora e altamente não formadora (CHOU; FASMAN, 1978).

Entretanto, ainda não foram encontrados campos de força que reproduzam a tendência natural que a maioria das sequências de proteínas apresenta, que é a preferência por formas estruturais helicoidais ou estendidas.

Assim, uma alternativa tem sido considerar apenas *fragmentos* da estrutura secundária. Baseados nesta ideia, Baker e colaboradores desenvolveram o *software* ROSETTA (2015), obtendo grande sucesso para os alvos do tipo *free modelling* nos experimentos do CASP, tornando a montagem por fragmentos popular neste campo de pesquisa. Uma das grandes vantagens de se usar fragmentos é a possibilidade de uma redução significativa da entropia do espaço conformacional de busca. Foi demonstrado que a abordagem *knowledge-based* obteve mais sucesso na modelagem *ab initio* para a predição de estruturas de proteínas (SIMONS et al., 1997).

3.2.3.2 Métodos de Busca

O *método de busca* e as *funções de energia* estão intimamente correlacionados. Métodos de busca rápidos que se baseiam nos potencias *physics-based*, como simulações de Monte Carlo e algoritmos genéticos, têm demonstrado ser muito promissores tanto para o PSP quanto para o refinamento (YANG, 2009).

Os métodos de busca são extremamente importantes na modelagem *ab initio*, pois para uma dada função de energia, o método de busca irá identificar estruturas que apresentem um mínimo global de energia, gerando uma classe de estruturas candidatas onde, posteriormente, o *modelo de seleção* selecionará a estrutura final. É muito comum o uso de Dinâmica Molecular e de Monte Carlo como parte integrante dos métodos empregados nas simulações de exploração do espaço conformacional de macromoléculas, a exemplo das proteínas.

Simulações de Monte Carlo

O método de busca mais popular é o **Simulated Annealing** (KIRKPATRICK; GELATT; VECCHI, 1983), uma vez que pode ser aplicado a qualquer tipo de problema de otimização. Basicamente, seu funcionamento consiste em executar um algoritmo de Monte Carlo Metropolis para gerar uma série de estados conformacionais de acordo com a distribuição de Boltzmann para uma dada temperatura, onde inicialmente executa uma simulação de Monte Carlo a altas temperaturas, seguido de uma série de simulações, em intervalos estabelecidos, à medida que a temperatura vai diminuindo.

Dinâmica Molecular

A Dinâmica Molecular é um método capaz de resolver as equações de movimento de Newton para um sistema composto de N átomos interagentes, onde em uma aproximação clássica:

$$m_i \frac{\partial^2 \vec{r_i}}{\partial t^2} = \vec{F_i} \quad i = 1, ..., N. \tag{4}$$

podendo conter de centenas a milhares de partículas. As forças são as derivadas negativas das **funções de energia potencial** $U(r_1, r_2, ..., r_N)$, dadas por:

$$\vec{F_i} = -\frac{\partial U(r_i)}{\partial \vec{r_i}} = -\nabla_i U(r_i)\,. \tag{5}$$

Este método tem sido muito utilizado no estudo dos caminhos de *folding* (DUAN; KOLLMAN, 1998) e para o refinamento de estruturas quando se tem modelos de baixa resolução. Embora seja muito importante no estudo do *folding* de proteínas, a Dinâmica Molecular não tem demonstrado muito sucesso na predição de estruturas, pois uma das razões é o alto custo computacional até mesmo para proteínas pequenas ($\approx$ 100 resíduos), fazendo com que a simulação demande bastante tempo.

Algoritmos Genéticos

Conformational Space Annealing (CSA) (LEE; SCHERAGA; RACKOVSKY, 1998) é um dos algoritmos genéticos de maior sucesso e tem sido aplicado em vários problemas de otimização. Emprega algoritmos de Monte Carlo Metropolis para localizar os mínimos locais de energia e o *annealing* para a busca no espaço conformacional. Primeiramente, é feita uma busca por todo o espaço conformacional de mínimos locais e, depois, encurta-se a busca para pequenas regiões de baixa energia à medida que a distância é reduzida. Neste caso, a distância desempenha o mesmo papel da temperatura para o *simulated annealing*, iniciando-se com uma grande distância para abranger várias conformações e depois vai reduzindo gradualmente.

Otimização Matemática

O αBB (α *branch and bound*) (KLEPEIS; FLOUDAS, 2003) é o único método de busca rigorosamente matemático, não utiliza nenhuma heurística ou modelos estocásticos diferentemente de todos os outros métodos. Entretanto, uma das desvantagens é que quando uma solução é encontrada, são geradas várias proteínas com muitos graus de liberdade.

3.2.3.3 Modelo de Seleção

O modelo de seleção de proteínas é a fase final deste processo, depois de serem geradas as estruturas candidatas, agora é preciso selecionar a estrutura que mais se aproxima do estado nativo. Os modelos de seleção de estruturas podem ser classificados em dois tipos: *energy based* e *free-energy based* (LEE; WU; ZHANG, 2015).

Energy based

Neste método, são criados diferentes tipos de potenciais para que seja possível identificar qual é o estado de menor energia ao final da predição. Geralmente, existem três tipos de funções *energy based* para a avaliação de estrutura e pontuação:

(i) *physics-based*, como por exemplo o CHARMM (2015) (LAZARIDIS; KARPLUS, 1999);

(ii) *knowledge-based* (SIPPL, 1990), sendo que estas já foram discutidas anteriormente;

(iii) e uma função de compatibilidade estrutura-sequência, descrevendo a compatibilidade entre a sequência alvo e um certo modelo de estrutura (LUTHY; BOWIE; EISENBERG, 1992).

Free-energy based

Neste caso, o modelo de energia livre de uma dada conformação ξ é dado por:

$$F(\xi) = -k_B\, T\, ln\, Z(\xi)\,, \tag{6}$$

sendo:

$$Z(\xi) = \int e^{-\beta U(\xi)}\, d\Omega\ , \tag{7}$$

onde $\beta = 1/k_B T$, k_B é a *constante de Boltzmann*, T é a temperatura, $U(\xi)$ é a energia potencial e $Z(\xi)$ é a *função de partição*, a qual é proporcional ao número de ocorrências das estruturas na vizinhança de ξ durante a simulação.

Como foi visto, os modelos de seleção são importantes na predição final da estrutura. Assim, tem despontado um novo campo de pesquisa denominado MQAP – *Model Quality Assessment Programs* – com a finalidade de avaliar a qualidade dos modelos propostos (FISCHER, 2006).

3.3 Considerações Finais

Sob o ponto de vista computacional, as proteínas podem ser representadas por meio de coordenadas internas ou cartesianas. Diversos *softwares* realizam a renderização e visualização destas estruturas moleculares por meio dos seus arquivos FASTA ou PDB, sendo que um dos mais utilizados em *web browsers* é o Jmol (2015).

Segundo Echenique (2007), os métodos *in silico* de predição de estruturas de proteínas podem ser divididos em três categorias: modelagem por homologia, *threading* e *ab initio*. Enfatizou-se, em especial, a modelagem computacional *ab initio*, pois este é o método adotado por este trabalho.

CAPÍTULO 4

Otimização Multiobjetivo

Geralmente a maior parte dos problemas reais da área de otimização exige que vários objetivos sejam determinados simultaneamente, o que usualmente gera soluções conflitantes, ou seja, não existe uma solução única que seja melhor do que todas as outras. Desta forma, deve-se buscar não uma solução, mas um conjunto de *soluções eficientes* que satisfaçam uma dada condição de equilíbrio para o problema proposto (COELLO, 2006). Os problemas desta natureza são chamados de Problema de Otimização Multiobjetivo (POMO), onde envolve a minimização (ou maximização) simultânea de um conjunto chamado de **vetor de funções objetivos** que satisfaça a certas condições de restrição.

Definição 2. *Seja p o número de funções objetivos, então o POMO pode ser formulado como:*

$$\begin{aligned} \textit{maximizar/minimizar} \quad & f(x) = \{f_1(x), f_2(x), \ldots, f_p(x)\}; \\ \textit{restrita a:} \quad & g_j(x) \geq 0,\ j = 1, \ldots, J; \\ & h_k(x) = 0,\ k = 1, \ldots, K; \\ & x_i^{(inf)} \leq x_i \leq x_i^{(sup)},\ i = 1, \ldots, n. \end{aligned} \tag{8}$$

onde $f(x) = \{f_1(x), f_2(x), \ldots, f_p(x)\}$ é um vetor de funções objetivos, g_j e h_k são as funções de restrição*, sendo J e K os respectivos números de restrições. Os valores x_i definem o espaço das variáveis $\mathbb{X}$, denominado de* ***espaço de decisão****, limitado por $x_i^{(inf)}$ e $x_i^{(sup)}$.*

Otimização Mono-Objetivo

Note que para $p = 1$, a Eq. (8) torna-se um problema comum de um único objetivo (mono-objetivo), ou seja, o ótimo corresponde às soluções extremas (mínimas ou máximas). Portanto, os problemas multiobjetivos são válidos apenas para $p > 1$.

Definição 3. *Uma solução x_i é dita factível se, e somente se, satisfizer todas as restrições g_j e h_k, caso contrário a solução não é factível (DEB, 2001). O conjunto de todas as soluções factíveis forma a* região factível*, também chamada de* ***espaço de busca****.*

Se todas as funções objetivos forem de minimização, neste caso, deseja-se encontrar pontos $x \in \mathbb{X}$, tal que $f(x) \in min\, f(\mathbb{X})$. Os pontos que satisfazem a essa condição são chamados de *solucões eficientes*:

Definição 4. *Uma solução $x^* \in \mathbb{X}$ é eficiente se não existe outro ponto $x \in \mathbb{X}$ tal que $f(x) \leq f(x^*)$ e $f(x) \neq f(x^*)$. Quando um ponto factível não satisfaz a essa condição, este é chamado de ponto ineficiente.*

A *eficiência* é um conceito equivalente ao de **Pareto-ótimo**, estritamente ligado ao conceito de **não-dominância**, ou seja, é o conjunto das soluções não dominadas em $\mathbb{X}$ (FONSECA; FLEMING, 1995). A imagem do conjunto de soluções eficientes, ou conjunto de Pareto-ótimo, é denominada de *fronteira eficiente*, conhecida na literatura como **fronteira de Pareto** (SAMPAIO, 2011). Uma solução eficiente não pode ser melhorada com relação a qualquer objetivo sem que cause uma piora em, pelo menos, algum outro objetivo. Portanto, define-se **dominância** como:

Definição 5. *Diz que uma solução factível x' **domina** outra solução factível x'', representado por $x' \preceq x''$, se, e somente se, $f_i(x') \leq f_i(x'')$ para $i = 1, \ldots, p$ e $f_i(x') < f_i(x'')$ para pelo menos uma função objetivo $f_i(x)$.*

Deste modo, pode-se formular o conjunto Pareto-ótimo $\mathcal{P}$ como:

Critério da Dominância (Pareto-ótimo)

$$\mathcal{P} = \left\{ x' \in \mathbb{X} \,|\, \nexists\, x'' \in \mathbb{X} : f(x'') \preceq f(x') \right\}. \tag{9}$$

A dominância é um critério que permite comparar a qualidade de duas soluções em problemas do tipo POMO. Segundo Deb (1998), **o conjunto não-dominado e a fronteira podem ser ótimos locais ou globais**.

Definição 6. *Um subconjunto $\mathcal{O}$ de $\mathbb{X}$ é denominado conjunto ótimo local em Pareto se, e somente se, todas as suas soluções são não-dominadas em relação a uma determinada vizinhança do espaço de decisão $\mathbb{X}$. A imagem deste conjunto no espaço de objetivos define uma região $\mathcal{O}'$ chamada de fronteira ótima local em Pareto.*

Definição 7. *Um subconjunto $\mathcal{C}$ de $\mathbb{X}$ é denominado conjunto ótimo global em Pareto se, e somente se, todas as suas soluções são não-dominadas em relação a quaisquer conjuntos ótimos locais $\mathcal{O}$ no espaço $\mathbb{X}$. A imagem deste conjunto no espaço de objetivos define uma região $\mathcal{C}'$ chamada de fronteira ótima global em Pareto.*

Como foi visto, no POMO é impossível adotar a solução de extremo (máximo ou mínimo) de apenas um dos objetivos, uma vez que os demais critérios também são relevantes ao problema, as soluções de extremo de um único objetivo exigem um *compromisso* nos demais objetivos, mas geralmente apenas uma solução será escolhida no final, denominada de **solução de melhor compromisso**. Então a razão entre a quantidade que deve ser aumentada de um objetivo para que seja diminuído outro objetivo é denominada de ***tradeoff***. Os *tradeoffs* e as *soluções eficientes* são informações importantes para que o **tomador de decisão** (decisor), ou um critério de decisão, possa escolher a solução de melhor compromisso.

4.1 Metas em Otimização Multiobjetivo

Em geral, três importantes metas devem ser concluídas em problemas do tipo POMO (DEB, 2001):

1. Obter um conjunto de soluções que esteja o mais próximo possível da fronteira de Pareto;
2. Encontrar um conjunto de soluções com maior diversidade possível;
3. Realizar as duas metas anteriores com a maior eficiência computacional possível.

A primeira meta é comum a todos os problemas de otimização, uma vez que soluções muito distantes da fronteira de Pareto são indesejáveis. A segunda meta é específica para cada tipo de problema. No caso da otimização multiobjetivo, o espaço de busca e o espaço de decisão devem conter pontos adequadamente distribuídos a fim de garantir a diversidade de soluções. Mas isto pode exigir um alto custo computacional, então é necessário que tais soluções sejam obtidas eficientemente (DEB; MOHAN; MISHRA, 2003).

4.2 Métodos de Otimização Multiobjetivo

A seguir, são apresentados os métodos "clássicos" de resolução de problemas de otimização multiobjetivo.

4.2.1 Classificação dos Métodos de Otimização Multiobjetivo

Segundo Horn (1997), na solução de problemas do tipo POMO existem dois cenários possíveis:

1. **Busca de soluções**: refere-se ao processo de otimização adotado para se obter o conjunto Pareto-ótimo de soluções;
2. **Tomada de decisões**: refere-se à escolha de um critério apropriado para selecionar uma solução do conjunto Pareto-ótimo, onde o *tomador de decisão* poderá ponderar entre as diferentes soluções conflitantes.

De acordo com Fonseca e Fleming (1995), os métodos de otimização multiobjetivo podem ser classificados em três categorias:

Método *a priori*: tomada de decisão antes da busca

Neste caso, a tomada de decisão ocorre antes da busca, em que previamente se tem alguma informação sobre o perfil de solução mais adequado ao problema, atribuindo elementos de preferência para os objetivos e redirecionando a busca para encontrar soluções com este perfil. Geralmente, dois modos de configuração de preferência são empregados:

(i) Combinar os objetivos em um único objetivo, explicitando a preferência através de **atribuição de pesos** para cada objetivo;

(ii) Classificação ordinal das preferências, em que o problema é resolvido considerando apenas o primeiro objetivo na *ordem de preferência* predefinida sem considerar os demais objetivos. A seguir, o problema é resolvido para o segundo objetivo, sendo que este fica sujeito à solução do objetivo anterior. Repete-se este processo até contemplar todos os objetivos da ordem de preferência.

Método *a posteriori*: tomada de decisão depois da busca

Busca-se encontrar o maior número possível de soluções, considerando que todos os objetivos têm a mesma relevância, para só depois selecionar a mais adequada ao problema. A principal desvantagem deste método é o alto custo computacional devido ao tempo gasto na busca, porém como neste caso a alteração das preferências não interfere no tempo de execução, este método é recomendado para problemas nos quais as preferências são relativas.

Método *interativo*: inserção progressiva de preferências

Neste método é feito um direcionamento da busca, em tempo de execução, para

regiões que contenham soluções mais adequadas. Este direcionamento é feita pelo tomador de decisão, em que antes de cada interação, pode-se definir as prioridades guiando a busca a partir de uma região de soluções conflitantes. Uma desvantagem é a constante intervenção de um decisor humano, o que pode ser inapropriado para problemas mais complexos.

4.2.2 Métodos Clássicos de Otimização Multiobjetivo

Os métodos clássicos de otimização multiobjetivo consistem na **escalarização** do problema, ou seja, *um problema de vários objetivos é transformado em um problema de apenas um objetivo* (COHON, 1978). Outros métodos que dispensam a escalarização também são empregados na resolução do POMO, mas não serão discutidos neste trabalho, como por exemplo: o método de descida do gradiente para problemas de otimização multiobjetivo e o método de direções viáveis (FLIEGE; SVAITER, 2000), método de Newton (FLIEGE; DRUMMOND; SVAITER, 2009) e o método Simplex Multiobjetivo (ZELENY, 1974).

Na literatura, os métodos clássicos de otimização multiobjetivo são geralmente divididos em três:

1. **Método dos pesos**: todas as funções objetivos são combinadas em uma única função objetivo, desta forma o problema original é transformado em um problema de um único objetivo, respeitando as restrições originais;

2. **Método ε-restrito**: consiste na otimização do objetivo mais importante sujeitando-se às condições de restrição dos outros objetivos (HAIMES; LASDON; WISMER, 1971);

3. **Programação por metas ou *goal programming***: fornece solução para problemas de decisão com múltiplas metas, geralmente conflitantes, onde o tomador de decisão especifica níveis de prioridade para os objetivos e quaisquer desvios desses níveis são minimizados. Neste caso, as metas são satisfeitas sequencialmente pelo algoritmo de solução. Em vez de minimizar ou maximizar a função objetivo, são minimizados os desvios entre as metas.

Neste trabalho será utilizada uma função objetivo composta por termos energéticos e estrutuais, sendo que para cada termo é atribuído um peso apropriado. Portanto, uma descrição detalhada do **método dos pesos** é feita a seguir.

4.2.2.1 O Método dos Pesos

Do inglês, *The Weighting Method*, também conhecido como a *média da soma ponderada*, consiste na transformação de um problema multiobjetivo para mono-objetivo

através da atribuição de pesos para cada objetivo, obtendo-se, assim, uma combinação linear entre eles. Portanto, o problema original transforma-se em um problema de um único objetivo respeitando as restrições originais. Este método serve para obter uma aproximação da fronteira eficiente e possui a vantagem de ser simples. Zadeh (1998) utilizou primeiramente o método para critérios de performance e otimização.

Dado um vetor de pesos $w \geq 0$, tal que $||w|| = 1$, então a Eq. (8) escalarizada transforma-se no seguinte problema $P(w)$:

$$\begin{aligned} P(w): \quad & min \sum_{k=1}^{p} w_k f_k(x); \\ restrita\ a: \quad & g_j(x) \geq 0,\ j = 1, \ldots, J; \\ & h_k(x) = 0,\ k = 1, \ldots, K; \\ & x_i^{(inf)} \leq x_i \leq x_i^{(sup)},\ i = 1, \ldots, n. \end{aligned} \tag{10}$$

onde cada solução ótima do problema $P(w)$ é também solução da Eq. (8). Se o vetor w for escolhido *a priori* pelo tomador de decisão, então a solução ótima também é a *solução de melhor compromisso*. O método dos pesos consiste, dessa forma, em resolver a Eq. (10) para vetores w distintos com o intuito de obter uma *aproximação da fronteira de Pareto* a partir das soluções ótimas encontradas até que os pontos estejam adequadamente distribuídos na fronteira.

Todavia, existem algumas desvantagens na aplicação deste método. Não há garantias de que haverá uma boa área de cobertura da fronteira de Pareto, uma vez que não é possível saber se as soluções encontradas estão bem distribuídas entre todas as outras soluções eficientes do problema, além da dificuldade de encontrar um vetor de pesos w tal que $P(w)$ tenha soluções ótimas (SAMPAIO, 2011).

4.2.3 Métodos Heurísticos de Otimização Multiobjetivo

As heurísticas são métodos de resolução de problemas (geralmente de otimização) em que se faz necessário ter informações prévias específicas acerca do assunto em questão, ou mesmo uma solução inicial aproximada antes de iniciar a sua execução. Formalmente, não existem garantias matemáticas de que a solução encontrada seja a melhor ou ótima, ou mesmo que se encontrará alguma solução afinal. Em contrapartida, as meta-heurísticas generalizam as heurísticas. Consistem de estratégias para guiar ou modificar outra heurística, com a finalidade de produzir uma gama maior de soluções do que aquelas normalmente geradas pelas buscas de otimizações locais (TALBI, 2009; GLOVER; LAGUNA, 1997).

Entretanto, inúmeros problemas reais exigem uma modelagem complexa com um nível de sofisticação que excede os recursos disponíveis de máquina para realizar uma

simulação, ou mesmo não é possível ou não se sabe como de fato modelar o problema, sendo necessário recorrer às heurísticas para obter informações importantes sobre o perfil de soluções que satisfazem, ao menos, certos critérios de condições de contorno, condições iniciais, entre outros.

Dentre as várias técnicas computacionais de heurísticas que têm sido empregadas em problemas deste tipo, destacam-se os Algoritmos Evolutivos (AEs) e, em especial, os Algoritmos Genéticos (AGs); uma classe de meta-heurísticas inspiradas na Teoria da Evolução de Darwin (1859) (FOGEL, 1994; FOGEL; OWENS; WALSH, 1966; HOLLAND, 1975; GOLDBERG, 1989). De acordo com Michalewicz e Schoenauer (1996), um AG procura por um equilíbrio entre o aproveitamento das melhores soluções e a exploração do **espaço de busca**. Denomina-se Algoritmos Evolutivos Multiobjetivos (MOEAs, do inglês *Multi-Objective Evolutionary Algorithms*) os AGs aplicados a problemas de otimização multiobjetivo. Dentre eles, destaca-se o **NSGA-II** (*Non-dominated Sorting Genetic Algorithm II*), um tipo baseado em ordenação elitista[6] por não-dominância (DEB et al., 2000). A principal vantagem do NSGA-II é a forma como é mantida a diversidade entre as soluções não-dominadas.

4.3 Otimização Multiobjetivo do PSP Aplicando Algoritmos Evolutivos

Neste trabalho serão utilizados os algoritmos evolutivos implementados no *framework* 2PG como técnica heurística de otimização multiobjetivo para o PSP (CUTELLO; NARZISI; NICOSIA, 2005). Os AEs, usualmente, possuem as seguintes características fundamentais:

1. Representação dos indivíduos;
2. Inicialização da população;
3. Função de avaliação (*fitness*);
4. Operadores de mutação e recombinação (*crossover*);
5. Seleção de indivíduos.

4.3.1 Representação dos Indivíduos

O modo como os indivíduos serão representados depende do problema em que o AE está sendo empregado, sendo comum representar por matrizes, grafos, valores

[6] O elitismo é um processo de construção de uma nova população em que os melhores indivíduos são selecionados para a próxima geração sem sofrer nenhuma alteração "genética".

discretos, entre outros. No caso do PSP, os indivíduos são representados pelo conjunto formado pelos ângulos diedros ϕ e ψ (*backbone*) e χ (cadeia lateral), os quais representam os parâmetros livres das proteínas (CUI; CHEN; WONG, 1998). O comprimento de ligação e os demais ângulos são considerados em seus valores ideias e mantidos constantes.

4.3.2 Inicialização da População

A inicialização da população (conjunto de indivíduos) pode ser feita de forma randômica (aleatória) ou com base em informações prévias específicas do problema, conhecida como *heurística*. Para reduzir o espaço amostral dos ângulos conformacionais (ângulos da cadeia principal e da lateral), os valores dos ângulos torcionais do *backbone* (cadeia principal) são restritos aos valores da base de dados CADB-2.0 – *Conformation Angles Data Base* – (MOHAN et al., 2005). Por sua vez, os ângulos torcionais da cadeia lateral (rotâmeros[7]) são dados pela a biblioteca Tuffery et al. (1991), pois permitem uma otimização rápida dos ângulos do espaço conformacional das cadeias laterais de aminoácidos para uma dada conformação do *backbone*.

4.3.3 Função de Avaliação (*fitness*)

A função de avaliação depende do problema em que o AE está sendo empregado e é a etapa que envolve o maior custo computacional. Ela é utilizada, sobretudo, para verificar a acurácia dos diferentes AEs empregados para resolver o mesmo problema. A função de *fitness* no caso dos POMOs (capítulo 4) configura a **função objetivo** do problema.

A literatura recomenda, para o problema do PSP, definir no máximo 3 objetivos para compor a função objetivo a fim de garantir uma boa acurácia dos MOEAs (ISHIBUCHI; TSUKAMOTO; NOJIMA, 2008). Isto porque a dominância se baseia não em soluções analíticas, mas em soluções que são melhores que outras comparando-se os objetivos. Com apenas 2 objetivos verifica-se que em várias soluções já não há dominância, ou seja, não é possível definir qual solução domina qual.

À medida que o número de objetivos crescem, muito mais soluções deixam de apresentar dominância, o que prejudica cada vez mais obter uma solução adequada para o problema. Com mais objetivos, passa-se a comparar cada vez mais coisas muito distintas, de modo que não é possível chegar a uma conclusão de qual solução é melhor que a outra.

4.3.4 Operadores Genéticos

Os AEs normalmente implementam os operadores genéticos de mutação e *crossover* (recombinação) responsáveis por obter novos indivíduos da população inicial. O operador de recombinação faz com que os filhos herdem algumas características dos pais

[7] Rotâmeros são geralmente definidos como conformações das cadeias laterais de energia mais baixa.

e a implementação depende de como os indivíduos foram representados. Já o operador de mutação implementa modificações aleatórias e serve como ferramenta para avaliar todo o espaço de busca. A cada nova aplicação dos operadores genéticos, faz-se necessário avaliar as alterações ocorridas. No caso do PSP, os novos indivíduos gerados são as diferentes novas conformações da proteína. Outros operadores além destes podem ser desenvolvidos a depender da aplicação.

4.3.5 Seleção de Indivíduos

A seleção de indivíduos é feita por uma estratégia previamente estabelecida, obedecendo ao princípio da Teoria da Evolução de Darwin (1859) em que os indivíduos mais adaptados sobrevivem. Entretanto, alguns AEs podem não descartar completamente os que não passaram no critério de seleção, existindo ainda uma pequena chance de que os piores indivíduos sejam selecionados também.

4.4 Considerações Finais

Os problemas de otimização geralmente envolvem a maximização ou minimização de alguma função. No caso de problemas mono-objetivos, apenas a análise de um único objetivo é de interesse. Entretanto, existe uma diversidade de situações em que é preciso tratar vários objetivos simultaneamente, então é preciso encarar o problema sob o ponto de vista multiobjetivo (POMO). Procura-se, deste modo, encontrar as melhores soluções para o problema proposto que atenda ao máximo os requisitos dos vários objetivos. Assim, o conjunto de Pareto-ótimo (Eq. 9) fornece as melhores soluções observando as restrições do problema.

Dentre os métodos de solução de problemas do tipo POMO, os métodos heurísticos têm apresentados excelentes resultados, em especial os algoritmos evolutivos. Deb (2001), Dejong (2006) mostraram que os métodos baseados em AEs são teórica e empiricamente robustos até mesmo em espaços complexos. Segundo Faccioli (2012, p. 5), "os algoritmos evolutivos (na prática) podem ser definidos como um método de busca de uma solução ótima a partir de uma população de soluções candidatas". A vantagem de uso dos AEs advém de sua capacidade de explorar o espaço de busca, encontrando as melhores soluções (MICHALEWICZ; SCHOENAUER, 1996), além de ser fácil a sua implementação em problemas de otimização mono-objetivo e multiobjetivo.

CAPÍTULO 5

ProtPred-Gromacs (2PG)

O **P**rot**P**red-**G**romacs (2PG) é um *framework* desenvolvido em decorrência dos trabalhos de Lima et al. (2006) e Faccioli et al. (2011) para investigar o problema da predição de estruturas terciárias de proteínas, provendo uma estrutura de dados para tratar de informações oriundas de aplicações em Biofísica e Bioinformática Estrutural. Em geral, os *frameworks* de Bioinformática Estrutural disponibilizam apenas um único algoritmo de predição, assim não é possível obter todas as informações necessárias acerca da proteína, sendo preciso trabalhar com mais de um *software* ao mesmo tempo.

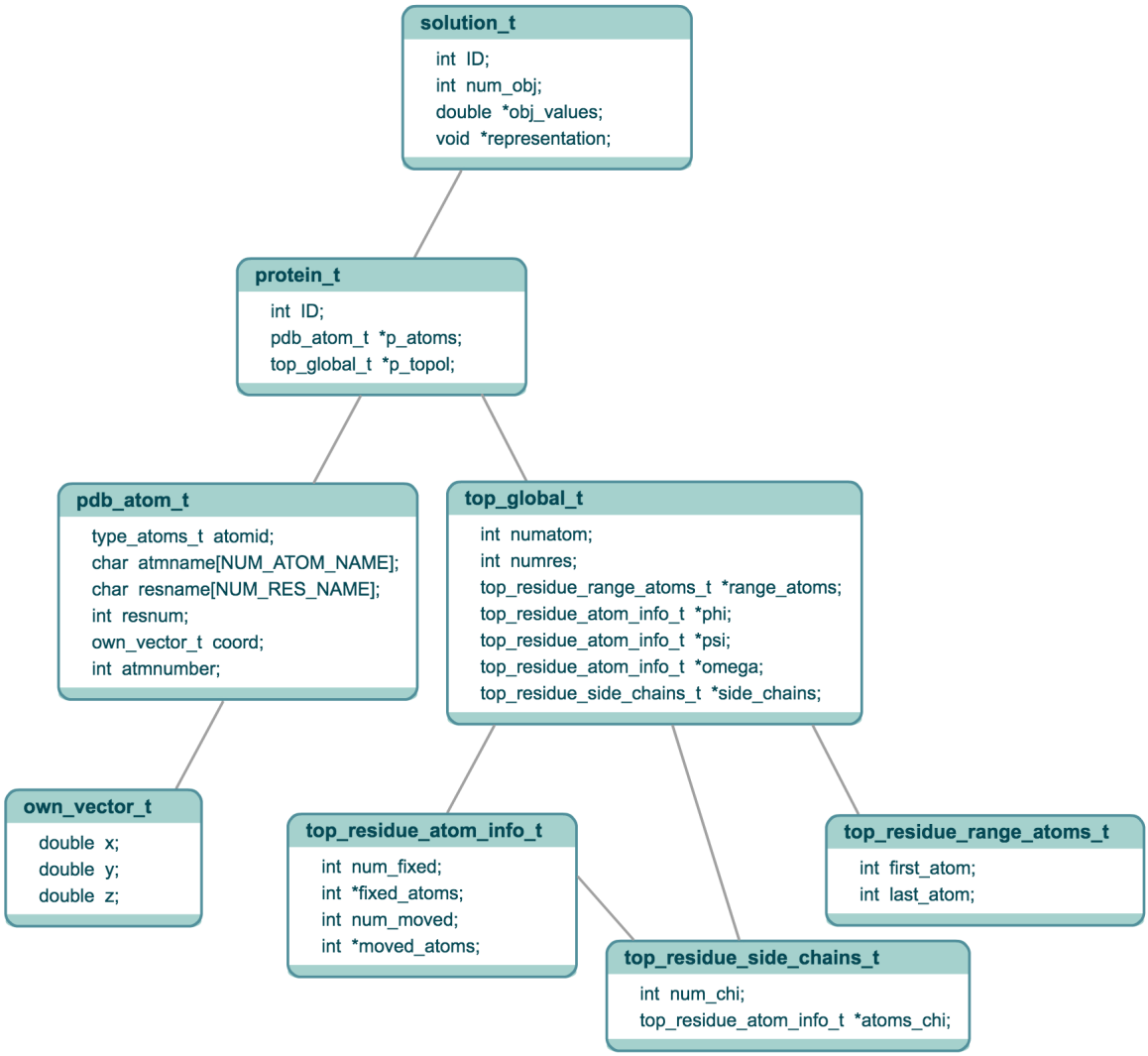
O 2PG surge para suprir essa carência, ele provê um único ambiente computacional para desenvolvimento e testes de metodologias integradas para investigar o PSP (*Protein Structure Prediction*). Possui ainda integração com o robusto *software* GROMACS para os cálculos das propriedades físicas das proteínas que são um dos objetivos (critérios) dos algoritmos. Dessa forma é possível modelar o *framework* para que o PSP seja tratado como um problema de otimização. Nesta abordagem, a técnica de otimização que o 2PG emprega faz uso de algoritmos evolutivos (AEs) mono-objetivos e multi-objetivos (MOEAs) para a predição de estruturas terciárias de proteínas.

O *framework* 2PG é um *software livre* escrito na linguagem de programação C e disponibilizado sob a licença Apache. O *framework* pode ser baixado no seguinte endereço eletrônico: <https://github.com/rodrigofaccioli/2pg_cartesian>, onde contém todas as instruções de instalação (FACCIOLI, 2016).

5.1 Estrutura de Dados do 2PG

A seguir, a Figura 37 ilustra a estrutura de dados do 2PG até a data de publicação deste trabalho. O tipo `solution_t` representa uma solução, ou seja, armazena valores necessários para modular uma solução, tais como a quantidade de objetivos <`num_obj`> a serem avaliados, um vetor <*`obj_values`> que armazena os valores de cada objetivo e <*`representation`> é um ponteiro para o tipo de representação de uma solução, que pode ser variada.

Figura 37 – Estrutura de dados do 2PG.



Fonte: adaptado de Faccioli (2015).

O 2PG representa a solução por meio da estrutura <`protein_t`>, onde <`pdb_atom_t`> é a representação atomística da proteína e <`top_global_t`> armazena a sua topologia (número de átomos e resíduos). As estruturas seguintes, <`pdb_atom_t`> e <`top_global_t`>, guardam as informações do arquivo *.pdb* (ver cap. 3, subseção 3.1.1) e da topologia global, respectivamente, sendo que <`own_vector_t`> representa a posição espacial de cada átomo da proteína. Para melhorar a performance de busca dos átomos pertecentes ao resíduo foi definido o tipo <`top_residue_range_atoms_t`>, enquanto que <`top_residue_atom_info_t`> permite um gerenciamento das gerações futuras de proteínas baseado em suas conformações, onde é possível rotacionar uma conformação selecionando valores para φ, ψ, ω e χ (Figura 24). Por fim, a última estrutura <`top_residue_side_chains_t`> trata das cadeias laterais por meio de informações relacionadas com os ângulos diedros χ.

5.2 Execução do 2PG

Para a execução do 2PG é necessário antes construir uma população inicial de conformações tridimensionais a partir da proteína-alvo. Isto é feito por meio do *software* `2pg_building_conformation`[8], onde é necessário apenas informar um arquivo de sequência primária que contém todos os resíduos da proteína-alvo (arquivo FASTA, ver cap. 3, subseção 3.1.1).

Estes resíduos precisam ser representados na forma atomística (*full-atom*) para a geração da população inicial, contendo informações a respeito dos comprimentos de ligação, ângulos de ligação e dos ângulos diedros (parâmetros livres) de cada átomo. Tais valores são obtidos a partir de sua topologia e diferentes conformações são geradas alterando-se os parâmetros livres com valores obtidos de uma biblioteca de ângulos diedros. A conformação passa por um processo final de minimização de energia fornecido pelo GROMACS, onde contatos indesejados entre os átomos possam ser removidos (ver subseção 5.3.1). Por fim, a população inicial está pronta para ser utilizada para iniciar o processo de predição.

O 2PG pode trabalhar com proteínas representadas tanto em **coordenadas internas** quanto em **coordenadas cartesianas** (ver cap. 3.1). Os algoritmos evolutivos utilizam as coordenadas internas para promover novas conformações estruturais devido à facilidade de seu uso, enquanto que os algoritmos de dinâmica molecular (GROMACS) utilizam a representação cartesiana para o cálculo das propriedades físicas, sem realizar nenhuma mudança estrutural. Portanto, em certo momento, é necessário utilizar um algoritmo de conversão entre os dois sistemas de coordenadas. Um dos mais utilizados é o `SN-Nerf` que já se encontra implementado no 2PG (PARSONS et al., 2005).

5.2.1 Operadores Genéticos do 2PG

Sob a perspectiva evolutiva, as novas conformações estruturais são obtidas por meio da definição dos chamados **operadores genéticos** que irão modificar os valores dos parâmetros livres. O *framework* 2PG implementa um operador de recombinação (*crossover*) e um operador de mutação.

Operador de *crossover*

Consiste do operador de *crossover* de um ponto, onde são utilizadas duas conformações. Escolhe-se aleatoriamente um certo resíduo da primeira conformação, depois todos os átomos da primeira conformação são copiados para uma nova conformação até o ponto onde foi escolhido aleatoriamente aquele resíduo. Então, para cada

[8] O *software* `2pg_building_conformation` pode ser baixado em: <https://github.com/rodrigofaccioli/2pg_build_conformation>.

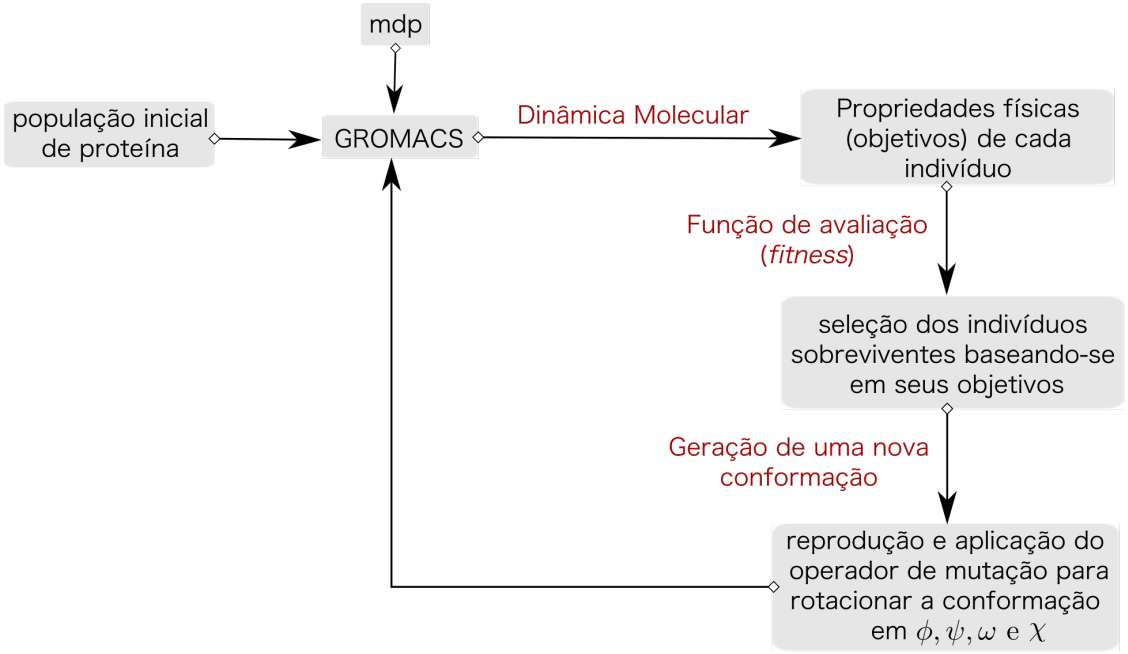
resíduo da nova conformação a partir do resíduo escolhido, são calculados os ângulos diedros ϕ, ψ, ω e χ tanto da primeira quanto da segunda conformação. Após, aplica-se a rotação com o valor da diferença de cada ângulo diétrico.

Operador de mutação

Utiliza-se apenas uma conformação para gerar uma nova conformação. Um resíduo é escolhido aleatoriamente e aplica-se alterações rotacionais em um de seus ângulos diedros. O valor da rotação também é escolhido de forma randômica, contudo, o *framework* permite determinar um intervalo de valores permitidos. É possível ainda escolher a quantidade de vezes com que se deseja aplicar rotações sucessivas.

A Figura 38 mostra o fluxo de funcionamento do 2PG ao receber uma população inicial de proteína:

Figura 38 – Fluxograma ilustrando as etapas de execução do 2PG.



Fonte: autor.

Para tornar o *framework* mais acessível e prático em sua execução, utiliza-se um arquivo de configuração que contém todos os parâmetros (`param_mc_temp.txt`) necessários para o algoritmo, como por exemplo o tamanho da população, o número de gerações e as opções dos objetivos a serem avaliados. A Tabela 5 lista todos os parâmetros e os valores passados para a execução do 2PG. Para cada proteína analisada, os parâmetros modificados foram: <`titulo`>, a população inicial <`pop_ini`>, o número de objetivos <`obj`> e os tipos de objetivos <`obj`> no final do arquivo.

Tabela 5 – Exemplo de parâmetros de execução do 2PG.

Parâmetro	Valor	Parâmetro	Valor
<titulo>	1VII_MC_temp_309	<nep>	1
<Nini>	1	<minimizacao>	ener_implicit
<algoritmo>	MonteCarlo	<nt>	1
<obj>	1	<ger>	500
<ind>	1	<pop_ini>	pop_0_1.pdb
<force_field>	amber99sb-ildn	<rotamer_library>	cad_tuffery
<rot_mut_phi>	30	<rot_mut_psi>	30
<rot_mut_omega>	30	<rot_mut_side_chain>	30
<apply_crossover>	no	<Started_Generation>	-1
<How_Many_Rotation>	1	<Individual_Mutation_Rate>	0.25
<MonteCarloSteps>	80000	<FrequencyMC>	100
<TemperatureMC>	309	<cros_1_Point>	1
<obj>	Potential	—	—

Fonte: autor.

De acordo com os parâmetros do 2PG exibidos na Tabela 5, é possível ter agora uma visão geral dos conceitos teóricos apresentados ao longo de todo este trabalho, resumidos da seguinte forma:

Proteínas

Os ângulos diedros ϕ, ψ, ω e χ tem importância fundamental na determinação das estruturas das proteínas (cap. 2, subseção 2.1.2). No 2PG, os parâmetros <rot_mut_phi>, <rot_mut_psi>, <rot_mut_omega> e <rot_mut_side_chain> recebem os valores desses ângulos que irão rotacionar a proteína, no exemplo da Tabela 5, no intervalo de $-30°$ a $30°$.

Dinâmica Molecular

Antes de iniciar a simulação de Dinâmica Molecular (cap. 3, sub-subseção 3.2.3.2), a energia do sistema deve ser minimizada para eliminar "maus contatos" entre os átomos, constituindo uma forma de otimizar a geometria ao encontrar posições dos átomos que minimizem a energia potencial, relaxando as distorções nas ligações químicas, nos ângulos de ligação e nos contatos de van der Waals. Assim, o

parâmetro <`minimizacao`> emprega o método de minimização de energia implícita considerando a proteína imersa em um solvente implícito (cap. 6, seção 6.4). Nas simulações de Dinâmica Molecular, é necessário utilizar um campo de força <`force_field`> para os cálculos das propriedades físicas das proteínas, neste caso, é empregado o AMBER (LINDORFF-LARSEN et al., 2010).

Monte Carlo

Conforme já foi visto no capítulo 3 (sub-subseção 3.2.3.2), o <`algoritmo`> utilizado para a busca das estruturas será o Monte Carlo Metropolis e o Monte Carlo com Dominância. No capítulo 6 será visto que a acurácia do algoritmo de Monte Carlo (ver Figura 41) depende, sobremaneira, do número de passos <`MonteCarloSteps`> executados (cap. 6, seção 6.4). O parâmetro <`FrequencyMC`> determina com qual frequência serão salvas as soluções (os *models* no arquivo `.pdb`), ou seja, para o valor da Tabela 5, as estruturas serão salvas a cada 100 passos. Portanto, para 80000 passos, 800 estruturas serão salvas. O parâmetro <`TemperatureMC`> determina a temperatura de Monte Carlo (ver Eq. 24).

Função Objetivo

O *número de objetivos* <`obj`> a serem avaliados, neste exemplo, será apenas 1 (Eq. 8, para $p = 1$). Um outro parâmetro de mesmo nome <`obj`>, que fica no final do arquivo `param_mc_temp.txt`, seleciona qual *tipo de objetivo* (energia potencial) será utilizado.

Algoritmos Evolutivos

E, por fim, como foi visto neste capítulo (seção 4.3), o 2PG emprega algoritmos evolutivos para o problema do PSP (cap. 4, subseção 4.2.3). Os indivíduos são aqui representados pelas estruturas das proteínas, onde a quantidade de população inicial de indivíduos é dada pelo parâmetro <`pop_ini`>, o número de gerações é determinado por <`ger`>, <`apply_crossover`> determina se o operador de *crossover* será aplicado ou não. A taxa de mutação individual é dada por <`Individual_Mutation_Rate`>.

5.3 GROMACS

O GROMACS (2015a) – GROningen MAchine for Chemical Simulations – é um *software open-source* para realização de cálculos de alta performance em Dinâmica Molecular, ou seja, um método capaz de resolver as equações de movimento de Newton para um sistema composto de N átomos interagentes.

As equações são resolvidas em pequenos intervalos de tempo, o sistema então evolui durante determinado período mantendo as condições iniciais de temperatura e pressão, e as coordenadas são escritas em um arquivo de saída como função do tempo em intervalos regulares, representando assim a trajetória do sistema até que se atinja um estado de equilíbrio. Deste modo, realizando-se uma média sobre a trajetória de equilíbrio, várias propriedades podem ser extraídas.

Como já foi dito, as soluções obtidas na modelagem por AEs, no caso do PSP, são os tipos de conformações assumidas pela proteína. Contudo, para representar a conformação se faz necessário calcular antes as suas propriedades físicas (ou interações), pois são tais propriedades que consistem os objetivos a serem avaliados, como por exemplo, a energia potencial da proteína.

Deste modo, utiliza-se o GROMACS para realizar o cálculo das propriedades físicas da proteína e como ele trabalha com o uso de coordenas cartesianas, é preciso fazer a conversão das coordenadas internas para as cartesianas. O algoritmo utilizado para realizar a conversão é o SN-NeRF (*Self-Normalizing Natural Extension Reference Frame*), cuja implementação está descrita no trabalho de Parsons et al. (2005).

O GROMACS é composto por aproximadamente 75 programas executáveis, sendo que a maioria deles são ferramentas de análise para os dados da trajetória e energias geradas nas simulações de Dinâmica Molecular (LINDAHL; HESS; SPOEL, 2001). Sua execução é via linha de comando no terminal, com uma interface simples para os arquivos de entrada e saída. A Tabela 6 mostra os arquivos e as extensões de arquivos que o GROMACS reconhece e utiliza internamente.

Tabela 6 – Tipos de arquivos do GROMACS.

Nome e Extensão (padrão)	**Tipo**	**Opção (padrão)**	**Descrição**
`atomtp.atp`	Asc	—	arquivo *atom type* usado por `pdb2gmx`
`eiwit.brk`	Asc	`-f`	arquivo *Brookhaven data bank*
`state.cpt`	xdr	—	arquivo *checkpoint*
`nnnice.dat`	Asc	—	arquivo de dados genérico
`user.dlg`	Asc	—	dados de *Dialog Box* para `ngmx`
`sam.edi`	Asc	—	*ED sampling input*
`sam.edo`	Asc	—	*ED sampling output*
`ener.edr`	—	—	energia genérica: `.edr, .ene`
`ener.edr`	xdr	—	arquivo de energia no formato portável `xdr`
`ener.edr`	Bin	—	arquivo de energia
`eiwit.ent`	Asc	-f	entrada no *Protein Data Bank*

Continua na próxima página...

Tabela 6 – continuação da página anterior.

Nome e Extensão (padrão)	Tipo	Opção (padrão)	Descrição
`plot.eps`	Asc	—	arquivo *Encapsulated PostScript* (tm)
`conf.esp`	Asc	-c	arquivo de coordenadas no formato `ESPResSo`
`conf.g96`	Asc	-c	arquivo de coordenadas no formato `Gromos-96`
`conf.gro`	Asc	-c	arquivo de coordenadas no formato `Gromos-97`
`conf.gro`	—	-c	estrutura: `.gro, .g96, .pdb, .esp, .tpr, .tpb, .tpa`
`out.gro`	—	-o	estrutura: `.gro, .g96, .pdb, .esp`
`polar.hdb`	Asc	—	base de dados do hidrogênio
`topinc.itp`	Asc	—	arquivo de topologia de inclusão
`run.log`	Asc	-l	arquivo de logo
`ps.m2p`	Asc	—	arquivo de entrada para `mat2ps`
`ss.map`	Asc	—	arquivo que mapeia os dados da matriz para cores
`ss.mat`	Asc	—	arquivo de dados de matriz
`grompp.mdp`	Asc	-f	arquivo de entrada `grompp` com os parâmetros de Dinâmica Molecular
`hessian.mtx`	Bin	-m	matriz Hessiana
`index.ndx`	Asc	-n	arquivo *index*
`hello.out`	Asc	-o	arquivo de saída genérico
`eiwit.pdb`	Asc	-f	arquivo PDB
`residue.rtp`	Asc	—	arquivo *type residue* usado por `pdb2gmx`
`doc.tex`	Asc	-o	arquivo LaTeX
`topol.top`	Asc	-p	arquivo de topologia
`topol.tpb`	Bin	-s	arquivo binário de entrada
`topol.tpr`	—	-s	entrada de arquivo de execução genérico: `.tpr, .tpb, .tpa`
`topol.tpr`	—	-s	estrutura+massa(db): `.tpr, .tpb, .tpa, .gro, .g96, .pdb`
`topol.tpr`	xdr	-s	arquivo portável de execução de entrada `xdr`
`traj.trj`	Bin	—	arquivo de trajetória (arquitetura específica)
`traj.trr`	—	—	trajetória de alta precisão: `.trr, .trj, .cpt`

Continua na próxima página...

Tabela 6 – continuação da página anterior.

Nome e Extensão (padrão)	Tipo	Opção (padrão)	Descrição
`traj.trr`	xdr	—	trajetória no formato de arquivo portável `xdr`
`root.xmp`	Asc	—	arquivo de matriz `X PixMap` compatível
`traj.xtc`	—	-f	arquivo de trajetória de entrada: `.xtc, .trr, .trj, .cpt, .gro, .g96, .pdb`
`traj.xtc`	—	-f	arquivo de trajetória de saída: `.xtc, .trr, .trj, .gro, .g96, .pdb`
`traj.xtc`	xdr	—	arquivo de trajetória compactado (formato portável `xdr`)
`graph.xvg`	Asc	-o	arquivo `xvgr/xmgr`

Fonte: GROMACS (2015b).

Dentre estes formatos de arquivos, vale a pena explicar alguns a fim de entender o fluxograma de funcionamento do GROMACS (Figura 39):

Trajectory (.trr)

Arquivo de saída do programa `mdrun`, armazena informações sobre os dados da trajetória da simulação, tais como as coordenadas, velocidades, forças e energias.

Generic energy formats (.edr, .ene)

Guarda informações das energias durante a simulação e as energias de minimização.

Protein Data Bank (.pdb)

Formato de arquivo no padrão do *Protein Data Bank*, contém informações sobre a posição dos átomos na estrutura das moléculas (RCSB PROTEIN DATA BANK, 2015d).

GROMACS Molecular Structure (.gro)

Fornece informações sobre a estrutura molecular assim como o arquivo *.pdb*, entretanto a principal diferença é que o arquivo *.gro* também armazena as velocidades.

Portable format for trajectories (.xtc)

Contém os dados da trajetória em coordenadas cartesianas.

Run input file (.tpr)

Arquivo binário com informações sobre a topologia do sistema, parâmetros, coordenadas e velocidades usado como *input* para o início da simulação.

Molecular Dynamics Parameter (.mdp)

Formato de arquivo em que o usuário configura todo os parâmetros a serem utilizados na simulação ou na minimização da energia.

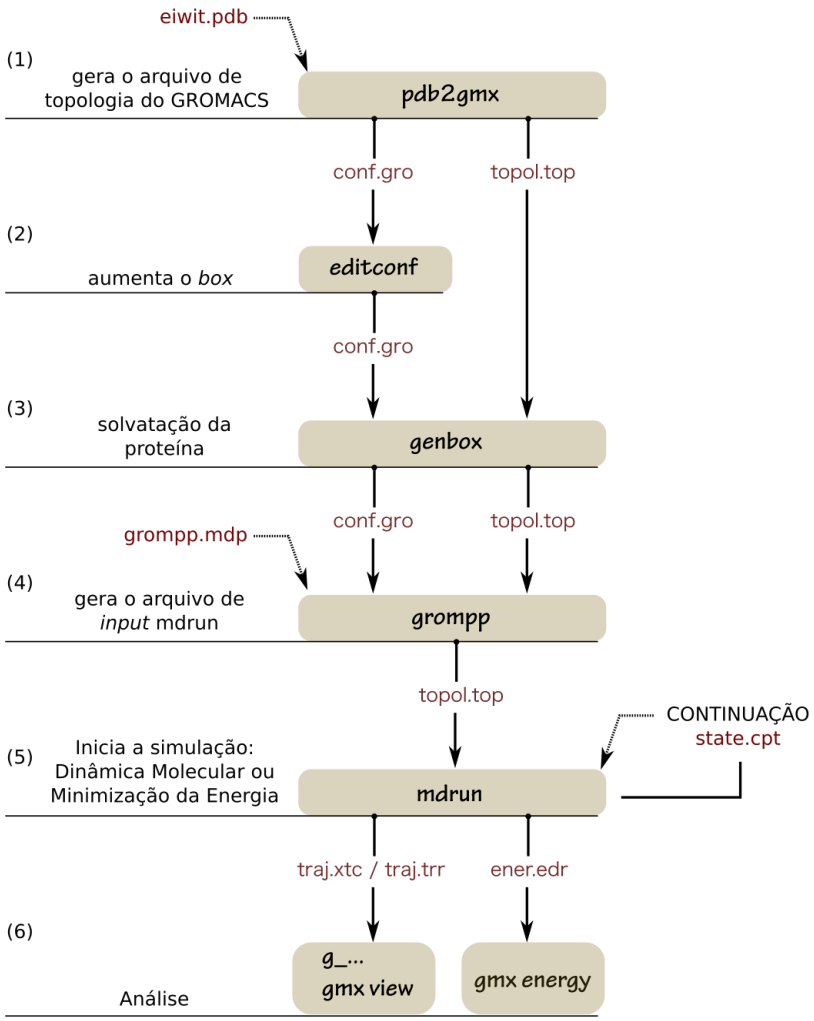
Checkpoint file (.cpt)

Formato de arquivo que contém o estado completo do sistema, necessário para que a simulação possa continuar.

5.3.1 Fluxograma de Funcionamento do GROMACS

A seguir, uma explicação das etapas do fluxograma de funcionamento do GROMACS está representado na Figura 39:

Figura 39 – Fluxograma de funcionamento do GROMACS.

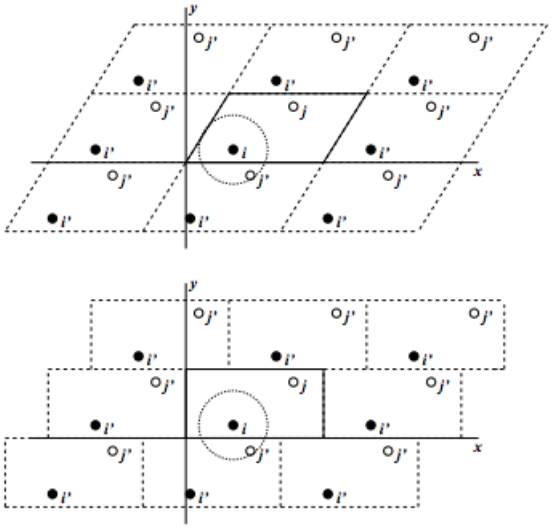


Fonte: adaptado de GROMACS (2015a).

(1) Conversão do arquivo *.pdb*:
O programa `pdb2gmx` converte o arquivo *.pdb* para o formato de leitura do GROMACS *.gro*, gerando também o arquivo de topologia *.top*.

(2) Criação do *box*:
A seguir, o programa `editconf` vai determinar o tamanho e o tipo de *box* (triclínico, cúbico ou octaédrico) que será utilizado na simulação. Isto porque o GROMACS utiliza um artifício clássico para minimizar os efeitos de borda em sistemas finitos, que é aplicar condições de contorno periódicas, colocando cada átomo do sistema em uma caixa (*box*), a qual é cercada por várias cópias transladadas de si mesma, como ilustra a Figura 40.

Figura 40 – Condições de contorno períodicas em duas dimensões utilizadas pelo GROMACS.



Fonte: GROMACS (2015a).

(3) Solvatação da proteína:
O próximo passo é solvatar a proteína inserida no *box* da etapa anterior com o programa `genbox`, que irá gerar o *box* definido pelo `editconf`.

(4) Minimização da energia:
O GROMACS utiliza o arquivo *.mdp* que deve ser inserido pelo usuário e que contém todos os parâmetros necessários para iniciar a simulação, então o programa `grompp` é acionado para gerar o arquivo de saída que será utilizado pelo programa `mdrun` para iniciar a minimização da energia.

(5) Simulação de Dinâmica Molecular:
O processo de simulação da Dinâmica Molecular é o mesmo que o da minimização da energia (etapa anterior), exceto por alguns parâmetros no arquivo

.mdp que não são usados na minimização da energia, como a opção de gerar a trajetória do sistema.

(6) Análise:
Depois de terminar a simulação, a etapa final é fazer uma análise da simulação com os seguintes programas:

a) `ngmx:` analisa a trajetória do sistema;

b) `g_energy`: monitora a energia;

c) `g_rms`: calcula o RMSD (Eq. 3), utilizando como medida a distância média entre os átomos das proteínas sobrepostas a fim de verificar a similaridade entre elas.

Alguns arquivos de *output* gerados pelo GROMACS, como o de trajetória e de coordenadas, requerem que seja feita uma renderização visual da estrutura molecular utilizando *softwares* externos, como os citados no capítulo 2 (subseção 3.1.2). Um análise da performance do GROMACS para simulações de Dinâmica Molecular de proteínas pode ser vista em Astuti e Mutiara (2009).

5.4 Considerações Finais

O 2PG investiga o problema de predição de estrutura de proteínas sob o ponto de vista de otimização, empregando técnicas de algoritmos evolutivos para obter soluções o mais próximo possível do estado nativo. A representação das proteínas pode ser feita por coordenadas internas ou cartesianas, entretanto, em virtude do GROMACS trabalhar em coordenadas cartesianas, é preciso fazer a conversão da matriz-Z.

Na representação das coordenas internas, torna-se muito fácil obter novas conformações alterando os valores dos ângulos diedros. Contudo, as coordenadas cartesianas trazem uma série de desvantagens em gerar diferentes conformações pela mudança da posição dos átomos. Em macromoléculas como proteínas, essas mudanças nas coordenadas podem modificar o comprimento de ligação com os átomos vizinhos, alterando, possivelmente, os ângulos torcionais e de ligação. Isto possibilita que pequenos erros surjam e o efeito acumulativo geraria resultados muito ruins na predição.

O 2PG e o GROMACS já implementam algoritmos que serão utilizados para avaliar os multiobjetivos deste trabalho. O GROMACS possui programas que calculam todas as propriedades estruturais (aSASA, pSASA e RG) e energéticas (potencial e GBSA) da função objetivo, enquanto que o 2PG já possui implementadas rotinas para a aplicação dos operadores genéticos e o cálculo da dominância.

CAPÍTULO 6

O Método de Monte Carlo

O método de Monte Carlo consiste em gerar, de forma aleatória, novas amostras a partir de um domínio de amostras que obedeça a uma dada função de distribuição de probabilidade. Este processo é repetido quantas vezes for necessário conforme a duração real do processo, ou em problemas em que se acredita que a distribuição seja estacionária, até que os novos valores gerados não apresentem mais mudanças de um passo a outro da simulação (METROPOLIS; ULAM, 1949).

O sistema, portanto, evolui de forma estocástica em razão da grande quantidade de números aleatórios gerados[9], contudo, as novas soluções numéricas obtidas são calculadas de forma determinística. Por exemplo, segundo Metropolis e Ulam (1949), imagine um sistema de muitas partículas onde cada partícula pode ser representada por um conjunto de valores, como as componentes dos seus vetores posição e velocidade, além de um índice para distingui-la das demais partículas.

Seja D_t o domínio inicial no instante t que, neste caso, é o conjunto de todas essas partículas antes da simulação. A simulação se inicia e por algum processo randômico novos valores de posição e velocidade são gerados para cada partícula, obtendo-se assim um novo domínio $D_{t+n\Delta t}$, onde n é uma fração do tempo total Δt gasto pela simulação, ou ainda, $n\Delta t$ é a duração de um passo (*step*) da simulação, uma vez que este processo é repetido várias e várias vezes. Portanto, pode-se calcular de forma determinística algumas propriedades do sistema como, por exemplo, o tempo médio gasto de cada partícula, uma vez que se tem vários conjuntos de valores de suas posições e velocidades.

Deste modo, o método de Monte Carlo é uma mistura de processos estocásticos e determinísticos, em que são obtidas soluções numéricas a partir de amostras randômicas, cujos resultados são computados de forma determinística dentro de um intervalo de aceitação estimado por tratamentos de erros estatísticos convencionais. Em geral, os métodos de Monte Carlo obedecem às seguintes etapas:

[9] Os números não são de fatos aleatórios, mas pseudo-aleatórios, pois nenhum processo computacional clássico conhecido gera números genuinamente randômicos.

1. Definir um domínio de variáveis de entrada iniciais que obedeçam a uma determinada função de distribuição de probabilidade;

2. Gerar aleatoriamente amostras a partir deste domínio, sendo que a frequência de distribuição de novas amostras seja a mesma que aquela que governa a mudança de cada parâmetro no domínio;

3. Computar, de forma determinística, os valores médios das propriedades desejadas;

4. Repetir os passos 2 e 3 durante o tempo necessário para cada tipo de problema, ou até convergir os valores, obtendo assim uma medida mais acurada do valor médio da propriedade de interesse;

5. Reunir e analisar os resultados gerados, fazendo os devidos tratamentos de erros estatísticos.

Uma condição necessária às simulações de Monte Carlo é assumir a *hipótese de ergocidade*, ou seja, todos os pontos do espaço de fase são igualmente prováveis de serem visitados se o algoritmo for executado por um longo período de tempo.

Existem várias áreas de aplicações do método de Monte Carlo, como por exemplo: física estatística, química, engenharia, biologia computacional, computação gráfica, mercado financeiro para análise de riscos, entre outros. Em especial, historicamente, as simulações com o método de Monte Carlo tiveram um papel importante no desenvolvimento da bomba atômica pelo *Projeto Manhattan*, onde os cientistas Ulam, von Neumann e Fermi consideraram o uso do método para estudar o coeficiente de difusão do nêutron em certos materiais.

6.1 O Algoritmo de Monte Carlo

O método de Monte Carlo, portanto, consiste em gerar aleatoriamente um conjunto de N estados ξ_1, ξ_2, ξ_3, ... , ξ_N, tal que:

$$\lim_{N\to\infty} \frac{N_\xi}{N} = P(\xi)\,, \tag{11}$$

onde N_ξ é o número de estados aceitos e $P(\xi)$ é alguma distribuição uniforme de probabilidade. O algoritmo geral do método de Monte Carlo é anunciado da seguinte forma:

1. **Passo 1**: escolher um estado inicial ξ_n $(n = 1, \ldots, N)$;

2. **Passo 2**: calcular a probabilidade de transição $(n \to n+1 = m)$ para um novo estado ξ_m, geralmente com configuração similar a ξ_n, dada por:

$$\pi_{mn} = \frac{P(\xi_m)}{P(\xi_n)} \ . \tag{12}$$

Escolher um número randômico ζ com valor entre 0 e 1. Então faça:

$$\begin{aligned} &\xi_{n+1} = \xi_m \,, \quad \text{para } \zeta < \pi_{mn} \quad \text{(muda para o novo estado).} \\ &\xi_{n+1} = \xi_n \,, \quad \text{caso contrário} \quad \text{(permanece no mesmo estado);} \end{aligned} \tag{13}$$

3. **Passo 3**: repetir o **passo 2**, substituindo ξ_n por ξ_{n+1}. O **passo 3** é repetido M vezes, sendo que M é um número suficientemente grande. Assim, de acordo com o **Passo 2**, a probabilidade de realizar a transição entre os estados pode ser resumido como:

$$\pi_{mn} = \begin{cases} \dfrac{P(\xi_m)}{P(\xi_n)} \,, & \text{se } \zeta < \pi_{mn} \,. \\ \\ 1 \,, & \text{caso contrário.} \end{cases} \tag{14}$$

6.2 Simulações de Monte Carlo em Sistemas Moleculares

O método de Monte Carlo tem sido muito importante no estudo de biologia molecular estrutural, sendo normalmente utilizado de duas formas:

1. Estimar propriedades termodinâmicas do espaço de conformações (ZHANG; KIHARA; SKOLNICK, 2002) e, em alguns casos, propriedades cinéticas também (SHIMADA; SHAKHNOVICH, 2002);

2. Procurar por conformações de baixa energia, incluindo a estrutura nativa da proteína (estado de mais baixa energia) (ZHANG; SKOLNICK, 2001).

No contexto da simulação molecular, como no caso de proteínas, o método de Monte Carlo baseia-se na técnica estatística de *importance sampling*, que consiste em estimar valores médios de propriedades de um sistema que obedeça a uma certa função de distribuição de probabilidades.

Dada a configuração inicial do sistema, o método de Monte Carlo tenta realizar uma mudança na configuração das partículas, que pode ser aceita ou rejeitada por um **critério de aceitação**, o qual garante que as novas amostras obedeçam ainda a uma certa distribuição de probabilidade. Uma vez aceita ou rejeitada, é calculado o valor esperado de uma propriedade de interesse e, após várias repetições desses passos, é possível perfazer uma medida acurada do valor médio desta propriedade em questão (EARL; DEEM, 2008).

Em simulações de Dinâmica Molecular, a *distribuição de Boltzmann* é muito utilizada para o cálculo da energia média do sistema, dada por:

$$P(\xi) = \frac{1}{Z}\, e^{-\beta U(\xi)} \ , \tag{15}$$

com $\beta = 1/k_B T$, sendo k_B a *constante de Boltzmann*, T a temperatura, $U(\xi)$ é a energia potencial (normalmente expressa pela hamiltoniana do sistema) e Z é a *função de partição*.

Por exemplo, seja A uma variável randômica que representa alguma propriedade de interesse. Assim, o seu valor médio $\langle A \rangle$ é dado por:

$$\langle A \rangle = \frac{\displaystyle\int d\,\mathbf{\Gamma}^p\, e^{-\beta U(\mathbf{\Gamma}^p)} A(\mathbf{\Gamma}^p)}{\displaystyle\int d\,\mathbf{\Gamma}^p\, e^{-\beta U(\mathbf{\Gamma}^p)}} \ \pm \ \delta A \ , \tag{16}$$

onde $\mathbf{\Gamma}^p$ é a configuração de um sistema de p partículas (por exemplo, a posição das p partículas) e δA é o erro estatístico associado. Portanto, a densidade de probabilidade $\rho(\mathbf{\Gamma}^p)$ de encontrar o sistema na configuração $\mathbf{\Gamma}^p$ é:

$$\rho(\mathbf{\Gamma}^p) = \frac{e^{-\beta U(\mathbf{\Gamma}^p)}}{\displaystyle\int d\,\mathbf{\Gamma}^p\, e^{-\beta U(\mathbf{\Gamma}^p)}} \ . \tag{17}$$

Seja N o número total de novos pontos gerados aleatoriamente pelo método de Monte Carlo e que obedeçam à função de distribuição dada pela Eq. (17). Logo, a Eq. (16) pode ser aproximada como:

$$\langle A \rangle \approx \frac{1}{N} \sum_{n=1}^{N} A(\mathbf{\Gamma}_n^p) \ \pm \ \delta A \, . \tag{18}$$

Deste modo, o algoritmo de Monte Carlo gera vários estados não correlacionados entre si, ou seja, trata-se de uma cadeia de Markov. Neste caso, a nova configuração de estados não depende das configurações anteriores, a única dependência reside somente na configuração atual do sistema. Assim, se o sistema está no estado n, a probabilidade de transição para um estado m é definido como:

$$\pi_{mn} = \alpha_{mn} p_{mn} = \alpha_{mn} \frac{\rho_m}{\rho_n} \ , \tag{19}$$

onde π_{mn} é uma matriz de transição, α_{mn} é probabilidade de realizar uma mudança de estado, p_{mn} é a probabilidade de aceitar esta mudança e ρ é a densidade de probabilidade. Assumindo que α_{mn} seja simétrico, ou seja, $\alpha_{mn} = \alpha_{nm}$, Metropolis et al. (1953)

propuseram que o critério de seleção seja baseado nas variações de energia entre o novo sistema e o antigo, no que ficou conhecido como o *algoritmo de Monte Carlo Metropolis*.

6.2.1 O Algoritmo de Monte Carlo Metropolis

No algoritmo de Monte Carlo Metropolis, também conhecido como algoritmo de Metropolis-Hastings (HASTINGS, 1970), existem três possibilidades:

(1) Energia do novo estado menor que a do estado antigo ($\Delta E < 0$):

Se o novo estado m tem energia menor do que o estado antigo n, ou seja, $U(m) < U(n)$, então a mudança de estado é aceita definindo:

$$p_{mn} = 1 \quad \Longrightarrow \quad \pi_{mn} = \alpha_{mn} \ , \quad \text{para} \quad \rho_m \geq \rho_n \ . \tag{20}$$

(2) Energia do novo estado maior que a do estado antigo ($\Delta E > 0$):

Se o novo estado tem energia maior que o antigo, $U(m) > U(n)$, então a mudança apenas será aceita se:

$$p_{mn} = e^{-\beta[U(m)-U(n)]} > \zeta \ , \tag{21}$$

onde $\zeta \in [0, 1]$ é um número aleatório. Deste modo:

$$\pi_{mn} = \begin{cases} \alpha_{mn}\dfrac{\rho_m}{\rho_n} \ , & \text{se } p_{mn} > \zeta \, . \\ \\ 0 \, , & \text{caso contrário } (\alpha_{mn} = 0). \end{cases} \tag{22}$$

(3) Energia do novo estado igual a do estado antigo ($\Delta E = 0$):

Caso as energias do sistema novo e antigo sejam iguais, $U(n) = U(m)$, então a matriz de transição é dada por:

$$\pi_{mm} = 1 - \sum_{n \neq m} \pi_{mn} \ . \tag{23}$$

Critério de aceitação do Monte Carlo Metropolis

$$p_{mn} = min \left\{ 1, \, e^{-\frac{1}{k_B T}[U(m)-U(n)]} \right\} . \tag{24}$$

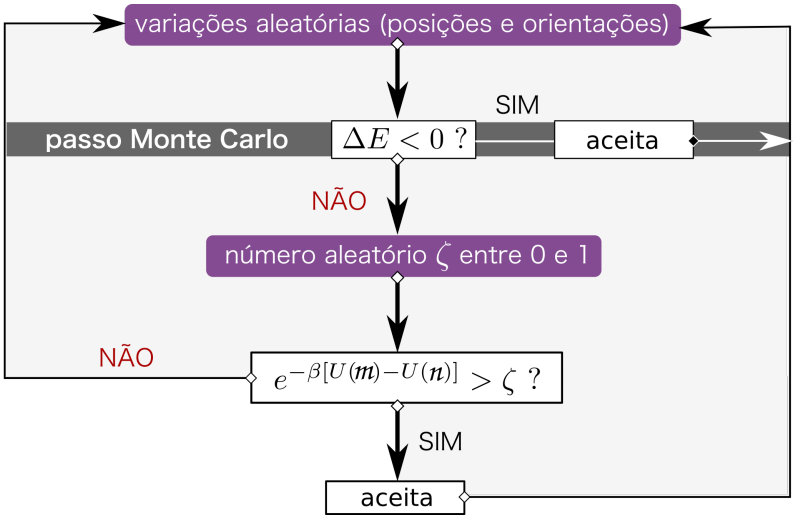
A Figura 41 representa o fluxograma do algoritmo de Monte Carlo Metropolis. Para o caso de sistemas moleculares, o método de Monte Carlo realiza pequenas perturbações nos graus de liberdade da molécula (LOTAN; SCHWARZER; LATOMBE, 2003).

Por exemplo, uma possível escolha seria selecionar aleatoriamente um átomo i do espaço conformacional e efetuar mudanças em suas coordenadas cartesianas:

$$x_i^{new} = x_i^{old} + \Delta(\chi - 0.5)\,, \tag{25a}$$
$$y_i^{new} = y_i^{old} + \Delta(\chi - 0.5)\,, \tag{25b}$$
$$z_i^{new} = z_i^{old} + \Delta(\chi - 0.5)\,, \tag{25c}$$

onde χ é um número pseudoaleatório entre 0 e 1, diferente para cada eixo a cada tentativa de mudança, e Δ seleciona o máximo deslocamento. Depois do movimento do átomo, é calculada a nova energia que será aceita ou rejeitada de acordo com o critério de aceitação de Metropolis (EARL; DEEM, 2008).

Figura 41 – Fluxograma do algoritmo de Monte Carlo Metropolis.



Fonte: autor.

Porém, para macromoléculas como proteínas, as mudanças nas coordenadas dos átomos não resulta em um método muito eficiente. Tais mudanças nas coordenas modificam os comprimentos de ligação com os átomos vizinhos, o que pode alterar os ângulos torcionais e de ligação, assim talvez seja improvável que a mudança seja aceita.

Dessa forma, é muito comum alterar somente os ângulos diedros do *backbone* e da cadeia lateral. Como, em geral, os ângulos e comprimentos de ligação entre duas ligações químicas sucessivas são quase constantes ao longo de toda a conformação em temperatura ambiente (KHOKHLOV; GROSBERG; PANDE, 1994), é uma prática comum fazer com que os ângulos e comprimentos de ligação sejam mantidos fixos durante a simulação e assumir que o único grau de liberdade seja a rotação dos ângulos diedros (torcionais). Neste trabalho, **são mantidos fixos os ângulos de ligação e os comprimentos de ligação, variando apenas os ângulos torcionais**.

Em geral, durante a execução da simulação computacional, três mudanças ocorrem de modo frequente:

1. **Mudanças estruturais:** a cada passo da simulação, alterações estruturais são realizadas;
2. **Critério de aceitação:** a regra segundo a qual as novas conformações são aceitas ou rejeitadas;
3. **Função de energia:** uma *pontuação* é atribuída para cada conformação, onde normalmente se escolhe a própria energia interna da conformação para pontuar.

6.3 Monte Carlo com Dominância

Neste trabalho, a proposta do Monte Carlo com Dominância é o de substituir o critério energético de Metropolis (Eq. 24) pelo critério de dominância (Eq. 9) entre duas soluções, nomeadas de solução nova x' e solução atual x''. Cada solução consiste de um *array* (vetor) que armazena os valores da função objetivo $f(x)$ no caso multiobjetivo.

A Dominância é uma técnica utilizada para vários tipos de problemas de otimização (cap. 4), onde não é possível determinar a solução exata de um problema que, em geral, é muito complexo para ser modelado analiticamente. Então, o que se faz é tentar eleger quais são os critérios, ou objetivos, que contribuirão decisivamente para a solução esperada. E a depender do problema, pode ser de interesse avaliar apenas um único objetivo (mono-objetivo) ou um conjunto de objetivos (multiobjetivo) simultaneamente.

6.3.1 O Algoritmo de Monte Carlo com Dominância

Considere que $f(x')$ seja a função objetivo da solução nova x' e $f(x'')$ a função objetivo da solução atual x'', as etapas a seguir mostram a execução do algoritmo de Monte Carlo com Dominância, havendo três possibilidades:

(1) Solução nova domina a solução atual ($f(x') \preceq f(x'')$):

$f_i(x') \leq f_i(x'')$ para $i = 1, \ldots, K$ em pelo menos um objetivo e $f_i(x') < f_i(x'')$ para todos os outros, sendo K o número total de objetivos.

Neste caso, a solução atual recebe os valores da função objetivo da solução nova e o algoritmo segue para o próximo passo (*step*) da iteração de Monte Carlo, gerando a próxima solução nova com outros valores.

(2) Solução nova é dominada pela solução atual ($f(x'') \preceq f(x')$):

$f_i(x'') \leq f_i(x')$ para $i = 1, \ldots, K$ em pelo menos um objetivo e $f_i(x'') < f_i(x')$ para todos os outros.

A solução atual permance com o seu valor corrente e segue para o próximo passo de Monte Carlo, onde será gerada outra solução nova com diferentes valores para a função objetivo.

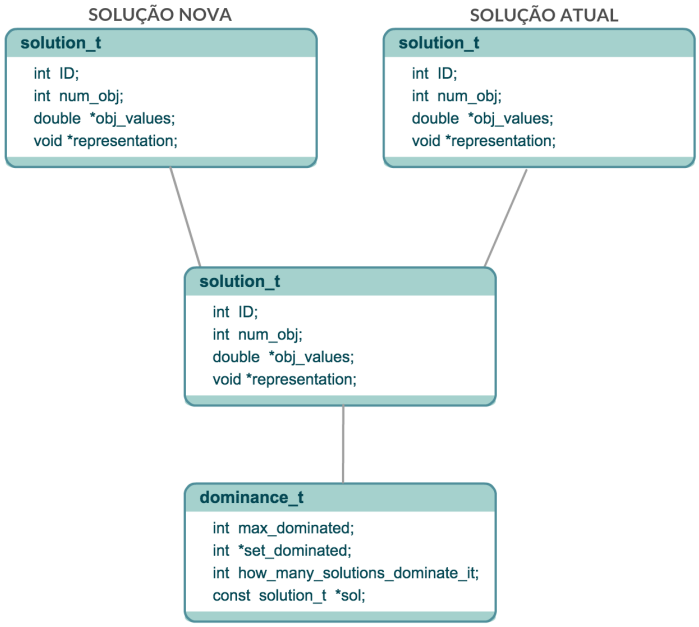
(3) Não há dominância:

Não existe nenhum objetivo $f_i(x)$ tal que o critério de dominância seja verificado, a solução atual permance inalterada e segue para o próximo passo de Monte Carlo gerando outra solução nova.

6.3.2 Implementação do Monte Carlo com Dominância no 2PG

A estrutura de dados para a implementação do Monte Carlo com Dominância é mostrada na Figura 42:

Figura 42 – Estrutura de dados do Monte Carlo com Dominância.



Fonte: autor.

Na estrutura `dominance_t`, <*`sol`> armazenará duas soluções, *solução nova* e *solução atual*, sobre as quais será verificada a dominância. Em <`max_dominated`> será guardado o número total de soluções dominadas, <*`set_dominated`> é um vetor contendo as soluções dominadas e <`how_many_solutions_dominate_it`> indica o número total de soluções que dominam a solução em questão.

Já em relação ao *framework* 2PG foram criados os seguintes arquivos:

`mc_dominance.h`

Arquivo de cabeçalho para declarar protótipos de funções em `mc_dominance.c` (recurso da linguagem C).

`mc_dominance.c`

Contém a implementação do método de Monte Carlo com Dominância, onde são definidas as estruturas da solução nova e atual (seção 6.3.3) para que sejam aplicadas as regras de dominância sobre os objetivos (seção 6.3) de cada solução. As regras do conceito de dominância já se encontram implementadas no *framework* 2PG no arquivo `dominance.c`.

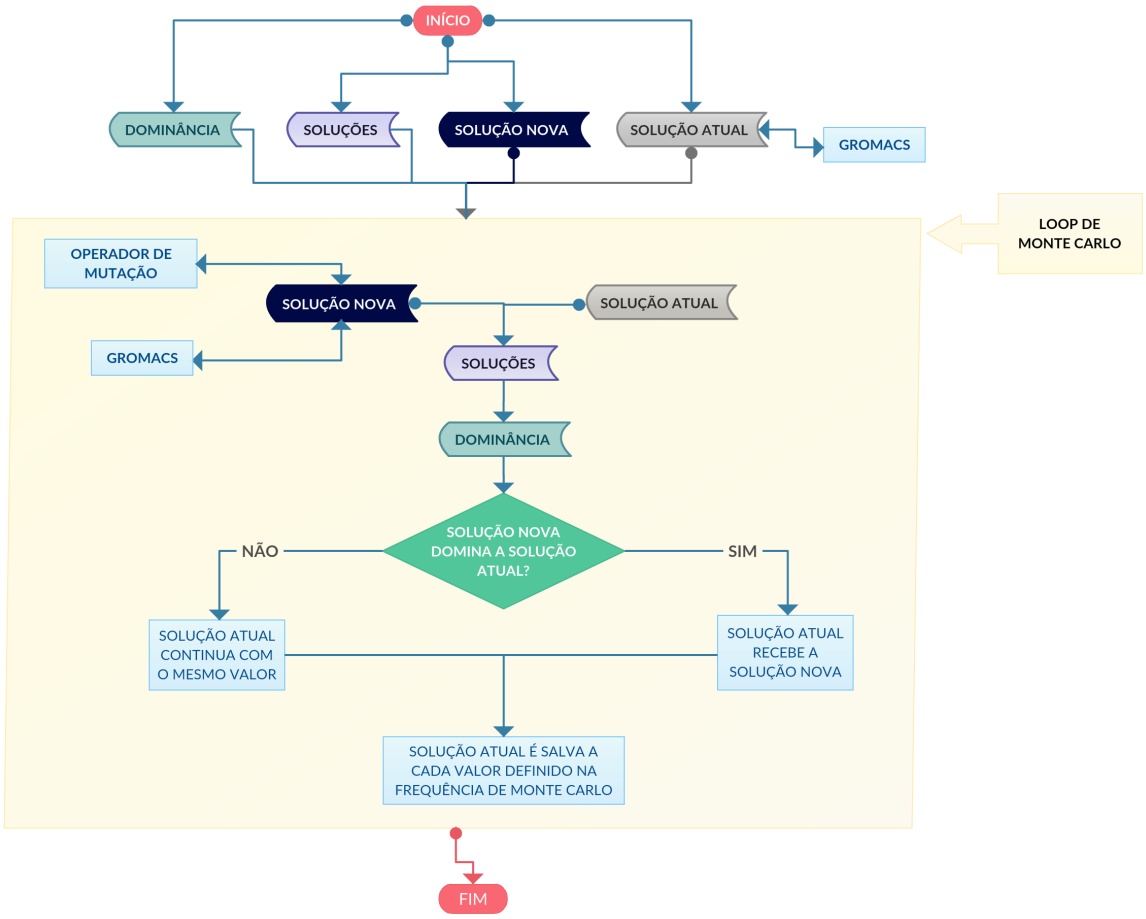
`protpred-Gromacs-MC_Dominance.c`

Inicializa a execução do `mc_dominance.c`.

6.3.3 Execução do Monte Carlo com Dominância no 2PG

Em princípio são criadas duas soluções, a **solução nova** e a **solução atual**. No início da execução de `mc_dominance` as duas são iguais e são calculados os objetivos da solução atual invocando o GROMACS. A seguir, o algoritmo inicia o *loop* dos passos de Monte Carlo. A Figura 43 esquematiza o funcionamento do algoritmo de Monte Carlo com Dominância.

Figura 43 – Fluxograma de execução do Monte Carlo com Dominância.



Fonte: autor.

Um operador de mutação altera os ângulos dos parâmetros livres da solução nova de forma randômica a fim de diferenciá-la da solução atual, e logo em seguida o GROMACS realiza os cálculos dos objetivos da solução nova. A aplicação do operador mutação consiste na etapa de escolha aleatória do espaço amostral característico do algoritmo de Monte Carlo (ver cap. 6, seção 6.1). As duas soluções são então reunidas em uma outra estrutura chamada **soluções**, a qual é passada para uma estrutura do tipo **dominância** necessária pelo programa `dominance.c`. A seguir, será verificado se a solução nova domina a solução atual, ou se a solução atual domina a solução nova, ou ainda se não houve dominância. Caso a solução nova domine a solução atual, a solução atual passa a receber a solução nova. De qualquer modo, no final, várias estruturas da solução atual são salvas em um arquivo *.pdb* a uma taxa definida pela frequência de Monte Carlo.

As várias estruturas salvas no arquivo *.pdb* de saída são conhecidas como **models**, configurando as estruturas preditas pelo algoritmo de Monte Carlo com Dominância. A saída deste aquivo *.pdb* é a última etapa de execução do programa `mc_dominance.c`. Os vários *models* são separados em arquivos *.pdb* individuais para o cálculo do RMSD de cada estrutura predita com a nativa. O próprio GROMACS já tem uma rotina implementada para o cálculo do RMSD por meio do programa `g_rms`.

6.3.4 Implementação das Funções Objetivos

Para este trabalho, o Monte Carlo com Dominância irá implementar as seguintes funções objetivos (*fitness*):

a) Energia potencial e energia de solvatação;

b) Energia potencial e área hidrofóbica;

c) Área hidrofóbica e área hidrofílica;

d) Raio de giro e área hidrofílica;

e) Raio de giro e energia de solvatação.

A seguir, serão apresentadas as definições energéticas e estruturais que o GROMACS aplica para o cálculo das propriedades físicas das proteínas. Convém salientar que o GROMACS implementa todas as rotinas necessárias para o cálculo destas propriedades, permitindo com que o 2PG as utilize como funções objetivos no tratamento de otimização multiobjetivo por meio dos algoritmos evolutivos (JAIMES; COELLO, 2008).

6.3.5 *Fitness* energético

Os objetivos energéticos considerados neste trabalho são relativos ao campo de força em que a proteína é submetida (energia potencial) e ao solvente em que ela está inserida (energia de solvatação).

6.3.5.1 Energia Potencial

Caso o objetivo seja a energia potencial, o GROMACS faz o seguinte cálculo para a energia potencial total do sistema:

$$\begin{aligned} \underset{energia\,potencial}{Objetivo} &= w_1 * E_{bond} + w_2 * E_{angle} + w_3 * E_{dihe} + w_4 * E_{imp} \\ &+ w_5 * E_{vdw} + w_6 * E_{elec} \ , \end{aligned} \tag{26}$$

em que w_i, com $i = 1, ..., 6$, representam os pesos a serem informados (ver Eq. 10) e os fatores E são as energias potenciais. Os quatro primeiros termos representam as ligações covalentes (energia de estiramento das ligações covalentes, de ângulos de torção, de Urey-Bradley e de Imprópria) e os dois últimos são os termos das ligações não-covalentes (van der Waals e eletrostática) (FACCIOLI, 2012).

A Eq. (26) utiliza parâmetros do campo de força AMBER gerenciado pelo GROMACS, então o 2PG não necessita de configurar tais parâmetros (LINDORFF-LARSEN et al., 2010). A energia potencial total dada pela Eq. (27a) consiste na soma das interações covalentes e não-covalentes (NAMBA; SILVA; SILVA, 2008; MACKERELL et al., 1998):

$$U_{total} = U_{bonded} + U_{non-bonded} \ , \tag{27a}$$

onde:

$$\begin{aligned} U_{bonded} = &\sum_{bounds} K_b(b - b_0)^2 \ + \ \sum_{angles} K_\theta(\theta - \theta_0)^2 \ + \ \sum_{UB} K_{UB}(S - S_0)^2 \\ &+ \ \sum_{impropers} K_{imp}(\varphi - \varphi_0)^2 \ + \ \sum_{dihedrals} \frac{V_n}{2}\big[1 + cos(n\chi - \delta)\big] \, , \end{aligned} \tag{27b}$$

$$U_{non-bonded} = \sum_{i,j} \left\{ \varepsilon_{ij} \left[\left(\frac{R_{min,ij}}{r_{ij}} \right)^{12} - 2 \left(\frac{R_{min,ij}}{r_{ij}} \right)^{6} \right] \ + \ \frac{q_i q_j}{4\pi\epsilon_0\epsilon_r r_{ij}} \right\} . \tag{27c}$$

Potenciais Harmônicos – Lei de Hooke

$$\sum_{bounds} K_b(b - b_0)^2 \ + \ \sum_{angle} K_\theta(\theta - \theta_0)^2 \ + \ \sum_{impropers} K_{imp}(\varphi - \varphi_0)^2 \ + \ \sum_{UB} K_{UB}(S - S_0)^2 \, .$$

O dois primeiros termos são devidos às oscilações dos comprimentos de ligação 'b e dos ângulos de ligação θ com relação aos seus valores de equilíbrio, respectivamente. O terceiro termo refere-se a um potencial torcional impróprio φ, responsável por

manter a estrutura tridimensional. E o último é o termo de Urey-Bradley, que consiste na interação baseada na distância S entre átomos que são separados por duas ligações consecutivas, denominada de interação-1,3. As variáveis K_b, K_θ, K_{imp} e K_{UB} são as constantes elásticas de cada termo. A aproximação harmônica é válida apenas para pequenas variações em relação aos valores de equilíbrio.

Energia Potencial de Torção

$$\sum_{dihedrals} \frac{V_n}{2}\big[1 + cos(n\chi - \delta)\big].$$

A energia potencial para uma torção é dado pelo termo diédrico, onde V_n é a barreira de energia para torção, n é o número de máximos (ou mínimos) de energia em uma torção completa, χ é o ângulo diedro e δ é o ângulo de fase.

Potencial de Lennard-Jones

$$\sum_{i,j} \varepsilon_{ij} \left[\left(\frac{R_{min,ij}}{r_{ij}} \right)^{12} - 2 \left(\frac{R_{min,ij}}{r_{ij}} \right)^{6} \right].$$

Os dois últimos termos da Eq. (27c) descrevem as interações entre pares de átomos (i, j) que não fazem ligação covalente (*nonbound*). No termo do potencial de Lennard-Jones, o parâmetro ε_{ij} é a profundidade do potencial entre a barreira atrativa e a repulsiva, e $R_{min,ij}$ é a distância (finita) em que o potencial entre as partículas é nulo. Ambos os parâmetros são ajustados experimentalmente ou por cálculos teóricos.

Potencial Eletrostático – Lei de Coulomb

$$\sum_{i,j} \frac{q_i q_j}{4\pi\epsilon_0\epsilon_r r_{ij}} \ .$$

O último termo é de natureza eletrostática, q_i e q_j são as magnitudes das cargas dos átomos i e j, r_{ij} é a distância entre as cargas, ϵ_0 é a permissividade do vácuo e ϵ_r é a constante dielétrica do meio.

6.3.5.2 Energia de Solvatação

A energia livre de solvatação é calculada utilizando o método GBSA (*Generalized Born Surface Area*), também conhecido como MM/GBSA (*Molecular Mechanics/GBSA*). Consiste de um método muito popular para o cálculo da energia mecânica molecular combinado com a superfície de acessibilidade do modelo de solvente implícito. Serve

para estimar a energia livre das interações entre pequenos ligantes de macromoléculas biológicas, utilizando algoritmos para obter soluções numéricas.

No método generalizado de Born, a energia livre de solvatação G_{solv} é dada por:

$$G_{solv} = G_{cav} + G_{vdw} + G_{pol} \ , \tag{28}$$

onde G_{cav} é o termo de interação solvente-solvente, G_{vdw} é o termo de interação de van der Waals soluto-solvente e G_{pol} é a interação eletrostática de polarização soluto-solvente.

A soma dos termos G_{cav} e G_{vdw} corresponde à energia livre de solvatação de uma molécula hidrofóbica da qual foram retiradas todas as cargas. Usualmente esta soma é denotada por G_{np}, envolvendo um cálculo que considera a área total de acessibilidade ao solvente apolar (aSASA) multiplicado pela tensão superficial. Logo, a Eq. (28) torna-se:

$$G_{solv} = G_{np} + G_{pol} \ . \tag{29}$$

6.3.6 *Fitness* estrutural

A seguir, uma breve explicação do significado de cada propriedade estrutural aplicada neste trablho:

Área Hidrofóbica

A região hidrofóbica de uma proteína situa-se em seu interior, formando um núcleo composto por aminoácidos apolares (ver Tabela 3) que tendem a repelir moléculas de água. Como foi demonstrado por Li, Tang e Wingreen (1997), o efeito hidrofóbico é a principal força indutora do *folding*. A área de acessibilidade ao solvente apolar é denominada **aSASA** (do inglês, *apolar Solvent-Accessible Surface Area*).

Área Hidrofílica

A região hidrofílica situa-se na superfície externa da proteína, composta pelos aminoácidos polares e eletricamente carregados. Estes estão em contato com o solvente devido à sua capacidade de formar ligações de hidrogênio. Frequentemente também interagem uns com os outros, formando as chamadas **pontes salinas**. A área de acessibilidade ao solvente polar é denominada **pSASA** (do inglês, *polar Solvent-Accessible Surface Area*). Tanto a área hidrofóbica quanto a hidrofílica são calculadas por meio de métodos numéricos específicos implementados no GROMACS.

Raio de Giro

O raio de giro (RG), ou *raio de giração*, é a distância a um ponto no qual se poderia concentrar a massa total (M) do corpo de modo que reproduziria o mesmo momento de inércia (I). Por definição, o raio de giro (RG) é dado por:

$$RG = \sqrt{\frac{I}{M}} \ . \tag{30}$$

No caso de partículas pontuais, o momento de inércia da i-ésima partícula é:

$$I_i = m_i ||\vec{r}_i||^2 \ . \tag{31}$$

Considerando que as partículas sejam os átomos da proteína, o raio de giro torna-se uma forma de avaliar o grau de compactação da estrutura (SPOEL et al., 2009). Deste modo, o raio de giro é dado por:

$$RG = \left(\frac{\sum_i ||\vec{r}_i||^2 \, m_i}{\sum_i m_i} \right)^{1/2} , \tag{32}$$

onde m_i é a massa do i-ésimo átomo e $\vec{r}_i$ é a sua posição em relação ao centro de massa da proteína.

O RG auxilia na verificação do estado de enovelamento da proteína, pois à medida que o colapso hidrofóbico ocorre, diferentes valores de RG em função do tempo indicam as etapas sucessivas do *folding*, uma vez que o vetor posição de cada átomo varia com o tempo $\vec{r} = \vec{r}(t)$.

6.4 Considerações Finais

O método de Monte Carlo permite estimar valores médios de propriedades de um sistema que segue uma determinada função de distribuição de probabilidade, em que amostras aleatórias do sistema são geradas a cada passo da simulação. A acurácia de uma determinada medida depende, sobremaneira, do número de passos dados na simulação computacional. A Dominância permite ao método de Monte Carlo considerar mais de um objetivo no processo de decisão de aceitação. Com a implementação do Monte Carlo com Dominância, espera-se verificar se os critérios estruturais também contribuem para o *folding* de proteínas, de modo a não haver uma dependência crucial dos parâmetros do campo de força adotado (ver Tabela 4). Tal análise será feira via cálculo do RMSD (ver Eq. 3).

Em geral, existe uma grande dificuldade em lidar com a proteína em solventes explícitos[10], pois como visto na análise das Eqs. (25), quaisquer mudanças nas coordenadas internas da proteína sem que também sejam alteradas as partículas do solvente, provavelmente irá resultar em uma sobreposição de átomos entre eles e afetar o critério de seleção. Portanto, simulações com solventes implícitos evitam este tipo de problema e são muito empregados nos métodos de Monte Carlo mais populares.

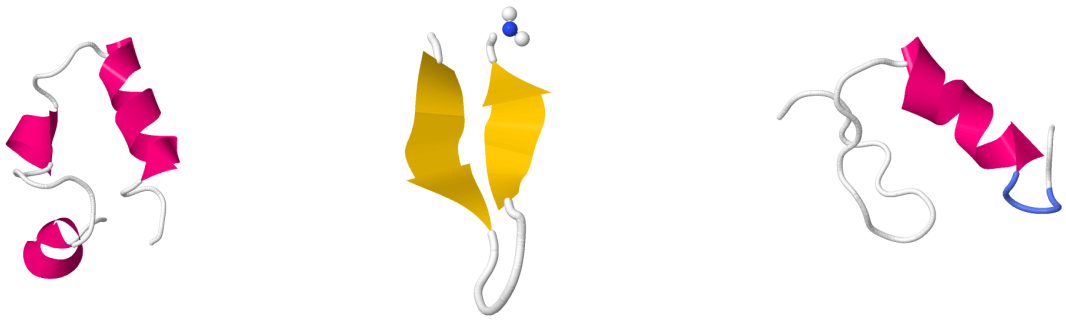
[10] Modelos de solventes explícitos consideram o solvente como sendo um meio discreto constituído de centenas a milhares de moléculas, consistindo em uma abordagem mais realística. Já modelos de solventes implícitos, consideram o meio como contínuo com propriedades que, na média, correspondem àquelas de um solvente real.

CAPÍTULO 7

Resultados & Análise

A aferição dos resultados obtidos baseia-se no cálculo de RMSD (Eq. 3) entre a proteína predita e a proteína-alvo, quanto menor o valor de RMSD, mais próxima a predição estará da estrutura nativa. As proteínas-alvo foram a 1VII, 1LE0 e 1FSD, conforme ilustra a Figura 44.

Figura 44 – Proteínas-alvo avaliadas neste trabalho.



(a) PDB ID: 1VII McKnight, Matsudaira e Kim (1997).

(b) PDB ID: 1LE0 Cochran, Skelton e Starovasnik (2001).

(c) PDB ID: 1FSD Dahiyat e Mayo (1997).

Fonte: RCSB Protein Data Bank (2015d).

Todos os valores de RMSD foram calculados entre as posições dos carbonos-alfa (C-α) da proteína predita e da proteína-alvo, de modo que a partir de agora esta consideração fica implícita nos resultados apresentados a seguir.

7.1 Predição das Estruturas Terciárias da 1VII, 1LE0 e 1FSD

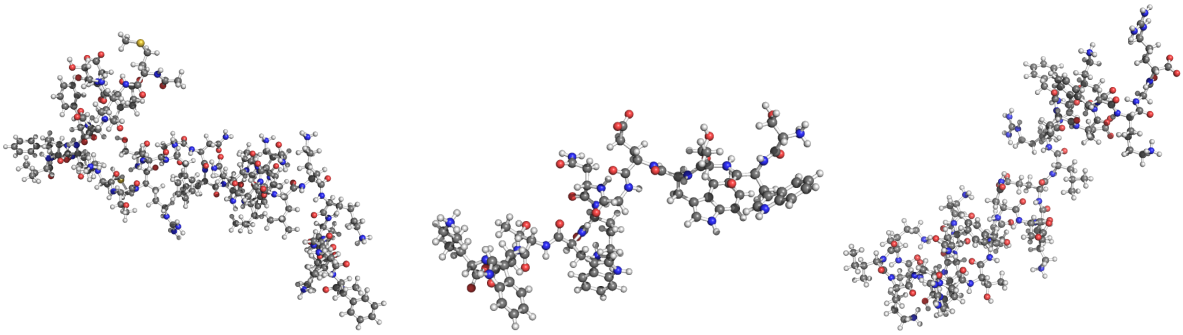
Inicialmente o programa `2pg_build_conformation` cria as populações iniciais inserindo apenas os arquivos FASTA de cada proteína-alvo (Tabela 7). As populações inciais criadas são representadas na forma *full-atom* (Figura 45).

Tabela 7 – Arquivos FASTA das proteínas 1VII, 1LE0 e 1FSD.

Proteína-alvo	**FASTA**
1VII	>1VII:A|PDBID|CHAIN|SEQUENCE XMLSDEDFKAVFGMTRSAFANLPLWKQQNLKKEKGLFX
1LE0	>1LE0:A|PDBID|CHAIN|SEQUENCE SWTWEGNKWTWK
1FSD	>1FSD:A|PDBID|CHAIN|SEQUENCE QQYTAKIKGRTFRNEKELRDFIEKFKGR

Fonte: autor.

Figura 45 – População inicial das proteínas 1VII, 1LE0 e 1FSD na representação *full-atom* criada pelo programa `2pg_build_conformation`.



(a) 1VII. (b) 1LE0. (c) 1FSD.

Fonte: autor.

Com a entrada da população inicial, o 2PG começa a executar a predição das estruturas terciárias para cada proteína. De acordo com o número de estruturas salvas dada pela frequência de Monte Carlo, foram geradas 800 estruturas e calculado os RMSDs entre cada estrutura predita e a sua respectiva proteína-alvo.

Por convenção, as estruturas preditas são identificadas (PID) por um número e são descritas como <`nativa`> PID <`num`>, onde <`nativa`> é a estrutura nativa da qual foi predita e <`num`> é o seu número de identificação (ID). Por exemplo, a proteína 1VII PID 50 refere-se à estrutura predita de número 50 (50 de 800 salvas), cuja estrutura nativa de origem é a proteína 1VII.

7.1.1 Refinamento Estrutural

Em geral, os algoritmos de predição de estruturas de proteínas geram conformações reduzidas, isto é, os aminoácidos são representados por um número reduzido de átomos para acelerar a procura de estruturas no espaço de busca. Portanto, esses modelos apresentam resoluções estruturais baixas no que diz respeito ao realismo físico da conformação (XU; ZHANG, 2011).

Em termos energéticos, os modelos reduzidos não são suficientes para a construção da topologia global da proteína. Algoritmos de refinamento estrutural auxiliam na recuperação dessas informações, normalmente comparando o modelo reduzido com a estrutura nativa. Para o refinamento dos resultados deste trabalho, foi utilizado um algoritmo de refinamento de minimização de energia a nível atômico chamado ModRefiner (2016). Este algoritmo funciona em duas etapas:

I. Construção da cadeia principal a partir dos C-α, considerando a topologia e a rede de ligações de hidrogênio;

II. Adição dos átomos da cadeia lateral à cadeia principal e otimização, empregando uma mistura de campo de força *physics-based* e *knowledge-based* (cap. 3, sub-subseção 3.2.3.1).

7.1.2 Configuração dos Testes

A Tabela 8 ilustra um exemplo das configurações dos parâmetros para rodar o Monte Carlo Metropolis:

Tabela 8 – Configuração de parâmetros de execução do 2PG para o Monte Carlo Metropolis.

Parâmetro	**Valor**	**Parâmetro**	**Valor**
<titulo>	1VII_MC_temp_309	<nep>	1
<Nini>	1	<minimizacao>	ener_implicit
<algoritmo>	MonteCarlo	<nt>	1
<obj>	1	<ger>	500
<ind>	1	<pop_ini>	pop_0_1.pdb
<force_field>	amber99sb-ildn	<rotamer_library>	cad_tuffery
<rot_mut_phi>	30	<rot_mut_psi>	30
<rot_mut_omega>	30	<rot_mut_side_chain>	30
<apply_crossover>	no	<Started_Generation>	-1
<How_Many_Rotation>	1	<Individual_Mutation_Rate>	0.25
<MonteCarloSteps>	80000	<FrequencyMC>	100
<TemperatureMC>	309	<cros_1_Point>	1
<obj>	Potential	—	—

Fonte: autor.

A Tabela 9 ilustra um exemplo da configuração de execução do 2PG para o Monte Carlo com Dominância aplicando a função objetivo raio de giro e área hidrofílica:

Tabela 9 – Exemplo de configuração de parâmetros de execução do 2PG para o Monte Carlo Dominância com a função objetivo raio de giro e área hidrofílica.

Parâmetro	**Valor**	**Parâmetro**	**Valor**
<titulo>	1VII_MC_temp_309	<nep>	1
<Nini>	1	<minimizacao>	ener_implicit
<algoritmo>	MC_Dominance	<nt>	1
<obj>	2	<ger>	500
<ind>	1	<pop_ini>	pop_0_1.pdb
<force_field>	amber99sb-ildn	<rotamer_library>	cad_tuffery
<rot_mut_phi>	30	<rot_mut_psi>	30
<rot_mut_omega>	30	<rot_mut_side_chain>	30
<apply_crossover>	no	<Started_Generation>	-1
<How_Many_Rotation>	1	<Individual_Mutation_Rate>	0.25
<MonteCarloSteps>	80000	<FrequencyMC>	100
<TemperatureMC>	309	<cros_1_Point>	1
<obj>	Gyrate Hydrophilic	—	—

Fonte: autor.

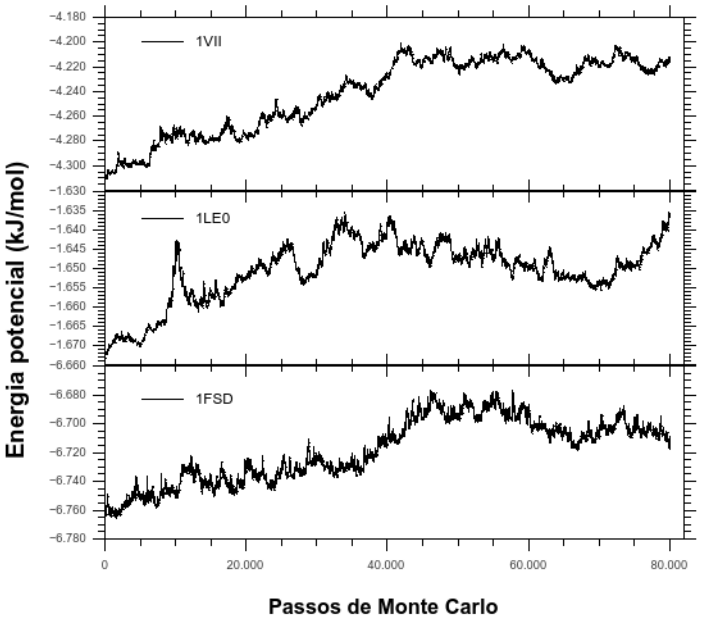
Convém ressaltar que em ambos os casos, Metropolis e Dominância, o número de passos de Monte Carlo executados foram de 80.000 e os ângulos diedros foram rotacionados (aleatoriamente) no intervalo de $-30°$ a $30°$.

7.1.3 Predição via Método de Monte Carlo Metropolis

Como já se sabe, a função objetivo do Monte Carlo Metropolis é sempre energia potencial, de modo que já fica subentendida esta consideração. O gráfico da Figura 46 mostra a variação da energia potencial em função dos passos de Monte Carlo.

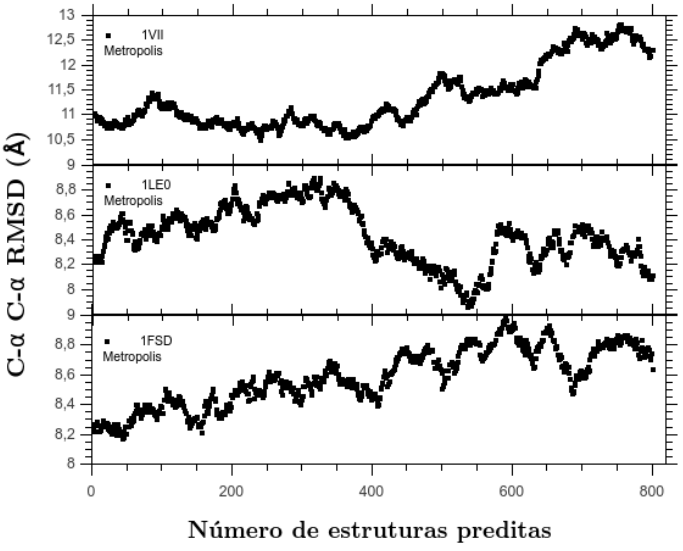
A Figura 47 mostra o gráfico do RMSD para as estruturas preditas. Uma vez que estamos interessados apenas nos valores mínimos de RMSD, o refinamento estrutural será aplicado somente para essas estruturas que mais se aproximam da conformação nativa.

Figura 46 – Perfil da energia potencial em função dos passos de Monte Carlo.



Fonte: autor.

Figura 47 – RMSD das 800 estruturas preditas com as suas respectivas proteínas-alvo via Monte Carlo Metropolis.



Fonte: autor.

A Tabela 10 mostra os valores de RMSD mínimo e máximo calculados das estruturas preditas, enquanto que a Tabela 11 mostra os valores de RMSD mínimos das estruturas refinadas, sendo que a 1LE0 PID 536 foi a que obteve o menor valor de RMSD para o algoritmo de Monte Carlo Metropolis. As Figuras 48 e 49 mostram as conformações estruturais refinadas de cada estrutura predita e o respectivo alinhamento com as suas proteínas nativas.

Tabela 10 – Valores de RMSD (mínimo e máximo) aplicando a função objetivo energia potencial no algoritmo de Monte Carlo Metropolis.

Proteína-alvo	**RMSD**$_{min}$ (Å)	**RMSD**$_{max}$ (Å)
1VII	10,491 (PID 239)	12,824 (PID 755)
1LE0	7,857 (PID 536)	8,906 (PID 315)
1FSD	8,169 (PID 44)	8,984 (PID 591)

Fonte: autor.

Tabela 11 – Valores de RMSD (mínimo) das estruturas refinadas (*ref*) aplicando a função objetivo energia potencial no algoritmo de Monte Carlo Metropolis.

Proteína-alvo	**RMSD**$^{ref}_{min}$ (Å)
1VII	8,080 (PID 239)
1LE0	2,066 (PID 536)
1FSD	4,223 (PID 44)

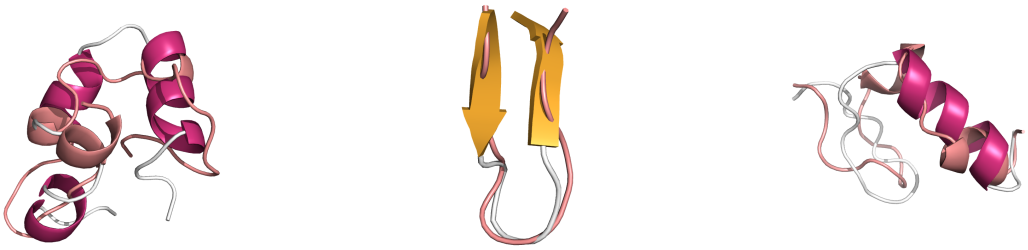
Fonte: autor.

Figura 48 – Conformação estrutural refinada das proteínas preditas aplicando a função objetivo energia potencial via Monte Carlo Metropolis.

(a) 1VII PID 40. (b) 1LE0 PID 536. (c) 1FSD PID 44.

Fonte: autor.

Figura 49 – Alinhamento das estruturas preditas *versus* estruturas nativas via Monte Carlo Metropolis.



(a) 1VII *vs.* 1VII PID 40. (b) 1LE0 *vs.* 1LE0 PID 536. (c) 1FSD *vs.* 1FSD PID 44.

Fonte: autor.

7.1.4 Predição via Método de Monte Carlo com Dominância

Para este trabalho foram avaliadas cinco funções objetivos (cap. 6, seção 6.3):

a) Energia potencial e energia de solvatação (GBSA);

b) Energia potencial e área hidrofóbica (aSASA);

c) Área hidrofóbica (aSASA) e área hidrofílica (pSASA);

d) Raio de giro (RG) e área hidrofílica (pSASA);

e) Raio de giro (RG) e energia de solvatação (GBSA).

A Tabela 12 mostra os resultados de RMSD para as cinco funções objetivos:

Tabela 12 – Valores de RMSD (mínimo e máximo) das cinco funções objetivos no algoritmo de Monte Carlo com Dominância.

	1VII		**1LE0**		**1FSD**	
Objetivos	**RMSD**$_{min}$ (Å)	**RMSD**$_{max}$ (Å)	**RMSD**$_{min}$ (Å)	**RMSD**$_{max}$ (Å)	**RMSD**$_{min}$ (Å)	**RMSD**$_{max}$ (Å)
RG-GBSA	9,686 (PID 223)	10,997 (PID 4)	4,194 (PID 478)	8,198 (PID 1)	6,654 (PID 714)	8,259 (PID 1)
RG-pSASA	10,949 (PID 25)	10,975 (PID 3)	8,154 (PID 42)	8,188 (PID 1)	8,258 (PID 723)	8,272 (PID 1)
aSASA-pSASA	10,978 (PID 12)	10,985 (PID 553)	8,219 (PID 5)	8,221 (PID 533)	8,289 (PID 8)	8,300 (PID 152)
Potencial-aSASA	10,980 (PID 1)	11,009 (PID 2)	8,204 (PID 1)	8,224 (PID 11)	8,271 (PID 6)	8,286 (PID 16)
Potencial-GBSA	11,017 (PID 1)	11,392 (PID 367)	8,214 (PID 28)	8,248 (PID 43)	8,270 (PID 2)	8,405 (PID 154)

Fonte: autor.

É possível notar que a função objetivo RG-GBSA resultou no menor valor de RMSD entre os objetivos considerados da dominância, como também no menor valor em relação ao Monte Carlo Metrópolis (Tabela 10), considerando-se este resultado para estruturas não refinadas. A seguir, serão analisados os gráficos das funções objetivos para as estruturas não refinadas, além do cálculo de RMSD e alinhamento estrutural após o processo de refinamento para cada uma das proteínas-alvo.

7.1.4.1 Raio de Giro e Energia de Solvatação

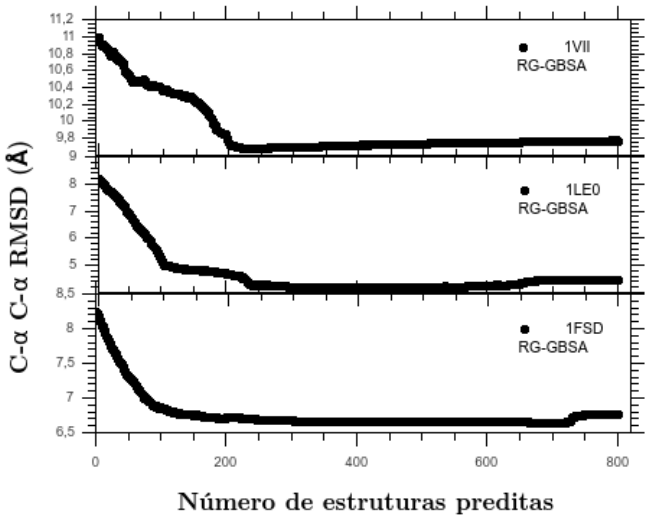
Tabela 13 – Valores de RMSD (mínimo) das estruturas refinadas (*ref*) aplicando a função objetivo RG-GBSA no algoritmo de Monte Carlo com Dominância.

Proteína-alvo	**RMSD**$^{ref}_{min}$ (Å)
1VII	6,932 (PID 223)
1LE0	3,450 (PID 478)
1FSD	4,451 (PID 714)

Fonte: autor.

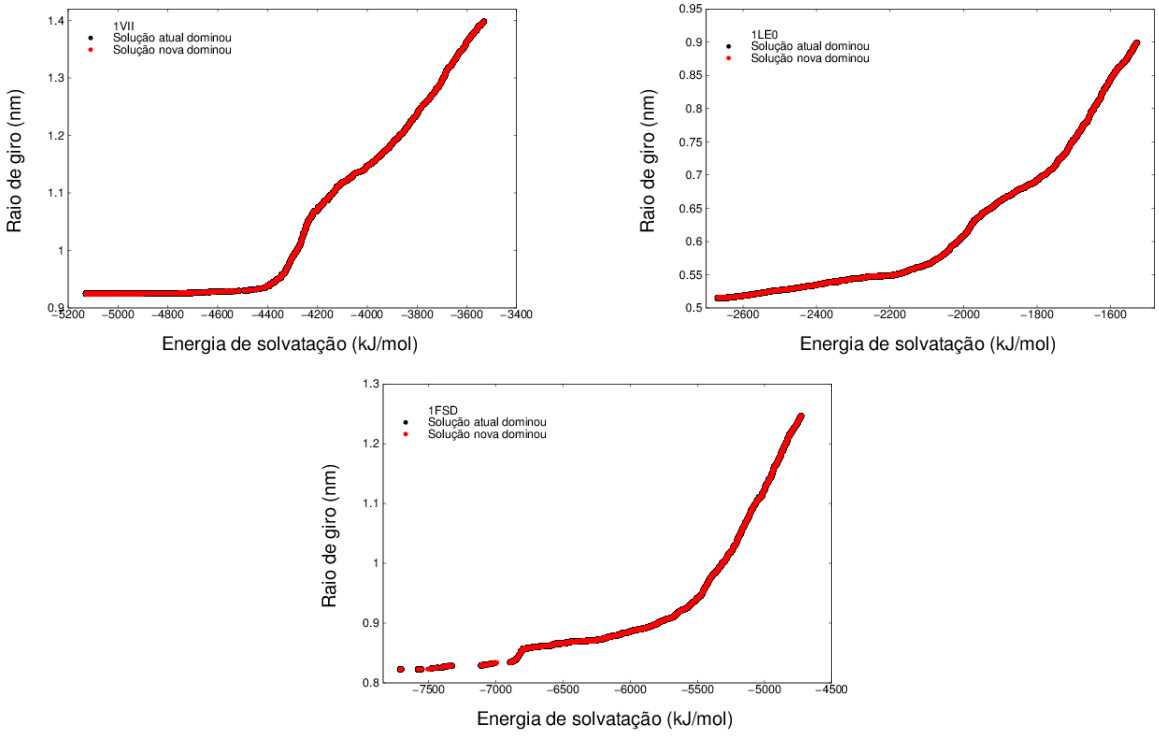
A Tabela 13 mostra os valores de RMSD (mínimo) para as proteínas preditas após o processo de refinamento estrutural. A Figura 50 mostra o gráfico de RMSD para todas as estruturas preditas não refinadas. A Figura 51 mostra a evolução das soluções nova e atual. Situações onde não há a dominância não são representadas nos gráficos.

Figura 50 – RMSD aplicando a função objetivo RG-GBSA no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

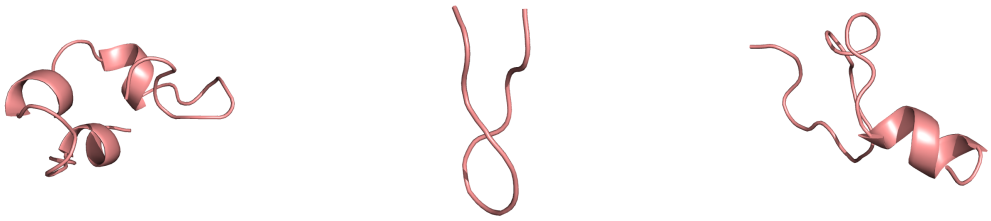
Figura 51 – Gráfico do raio de giro (RG) em função da energia de solvatação (GBSA) no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

Como é possível observar, a estrutura que apresentou o menor valor de RMSD foi a 1LE0 PID 478. As Figuras 52 e 53 mostram a conformações estruturais de cada estrutura predita refinada e o seu alinhamento com a proteína-alvo, respectivamente. Observa-se que na predição da 1LE0 não foi possível formar a estrutura de folha-β.

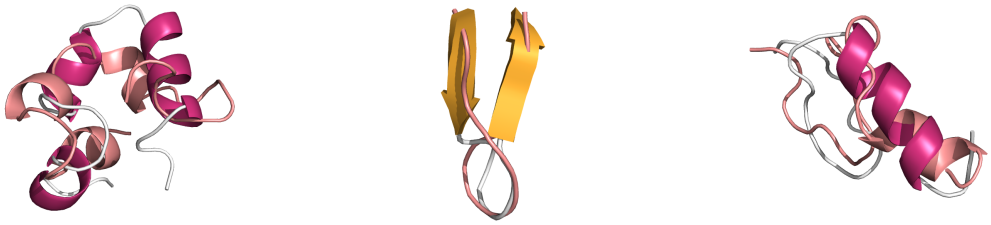
Figura 52 – Conformação estrutural refinada das proteínas preditas aplicando a função objetivo RG-GBSA via Monte Carlo com Dominância.



(a) 1VII PID 223. (b) 1LE0 PID 478. (c) 1FSD PID 714.

Fonte: autor.

Figura 53 – Alinhamento das estruturas preditas *versus* proteínas-alvo via Monte Carlo com Dominância.



(a) 1VII *vs.* 1VII PID 223. (b) 1LE0 *vs.* 1LE0 PID 478. (c) 1FSD *vs.* 1FSD PID 714.

Fonte: autor.

7.1.4.2 Raio de Giro e Área Hidrofílica

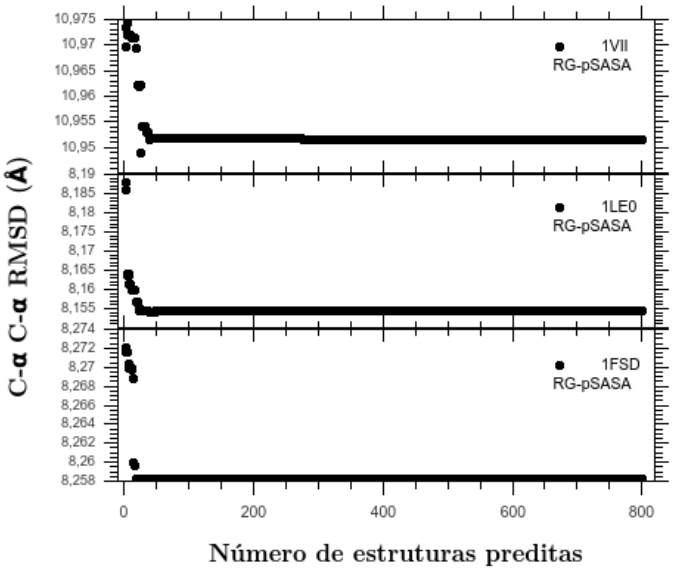
Tabela 14 – Valores de RMSD (mínimo) das estruturas refinadas (*ref*) aplicando a função objetivo RG-pSASA no algoritmo de Monte Carlo com Dominância.

Proteína-alvo	$\mathbf{RMSD}_{min}^{ref}$ (Å)
1VII	5,482 (PID 25)
1LE0	2,306 (PID 42)
1FSD	1,999 (PID 723)

Fonte: autor.

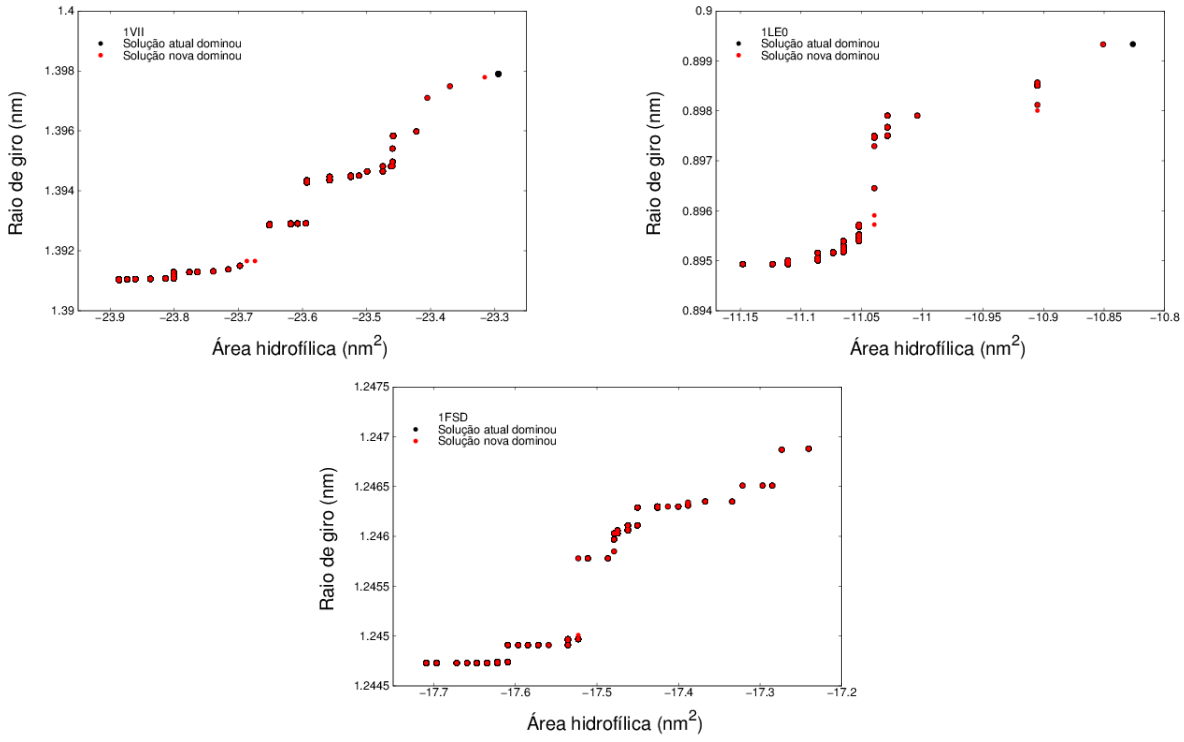
A Tabela 14 mostra os valores de RMSD das estruturas refinadas para a função objetivo RG-pSASA. A Figura 54 mostra o RMSD para as proteínas preditas não refinadas, enquanto que o gráfico da Figura 55 mostra o comportamento das soluções consideranto o raio de giro e a área hidrofílica.

Figura 54 – RMSD aplicando a função objetivo RG-pSASA no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

Figura 55 – Gráfico do raio de giro (RG) em função da área hidrofílica (pSASA) no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

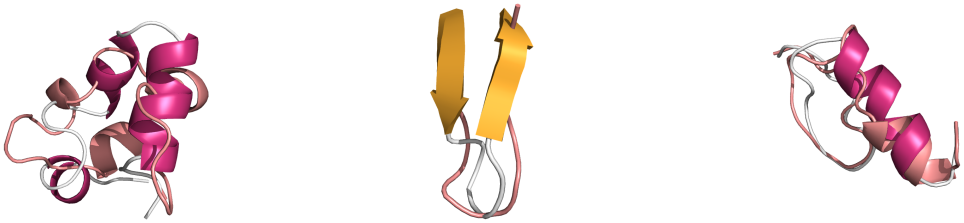
As Figuras 56 e 57 ilustram as conformações estruturais refinadas e o alinhamento com as suas proteínas-alvo, respectivamente.

Figura 56 – Conformação estrutural refinada das proteínas preditas aplicando a função objetivo RG-pSASA via Monte Carlo com Dominância.

(a) 1VII PID 25. (b) 1LE0 PID 42. (c) 1FSD PID 723.

Fonte: autor.

Figura 57 – Alinhamento das estruturas preditas *versus* proteínas-alvo via Monte Carlo com Dominância.



(a) 1VII *vs.* 1VII PID 25. (b) 1LE0 *vs.* 1LE0 PID 42. (c) 1FSD *vs.* 1FSD PID 723.

Fonte: autor.

7.1.4.3 Área Hidrofóbica e Área Hidrofílica

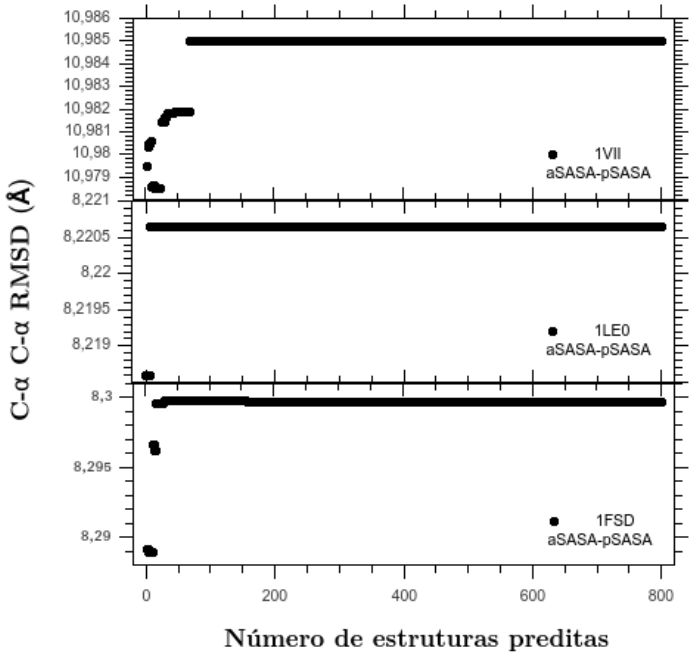
A Tabela 15 exibe os valores de RMSD calculados para as estruturas refinadas, enquanto que as Figuras 58 e 59 mostram o comportamento do RMSD das estruturas não refinadas e dos valores da função objetivo aSASA-pSASA, respectivamente.

Tabela 15 – Valores de RMSD (mínimo) das estruturas refinadas (*ref*) aplicando a função objetivo aSASA-pSASA no algoritmo de Monte Carlo com Dominância.

Proteína-alvo	$\mathbf{RMSD}_{min}^{ref}$ (Å)
1VII	5,691 (PID 12)
1LE0	2,474 (PID 5)
1FSD	3,988 (PID 8)

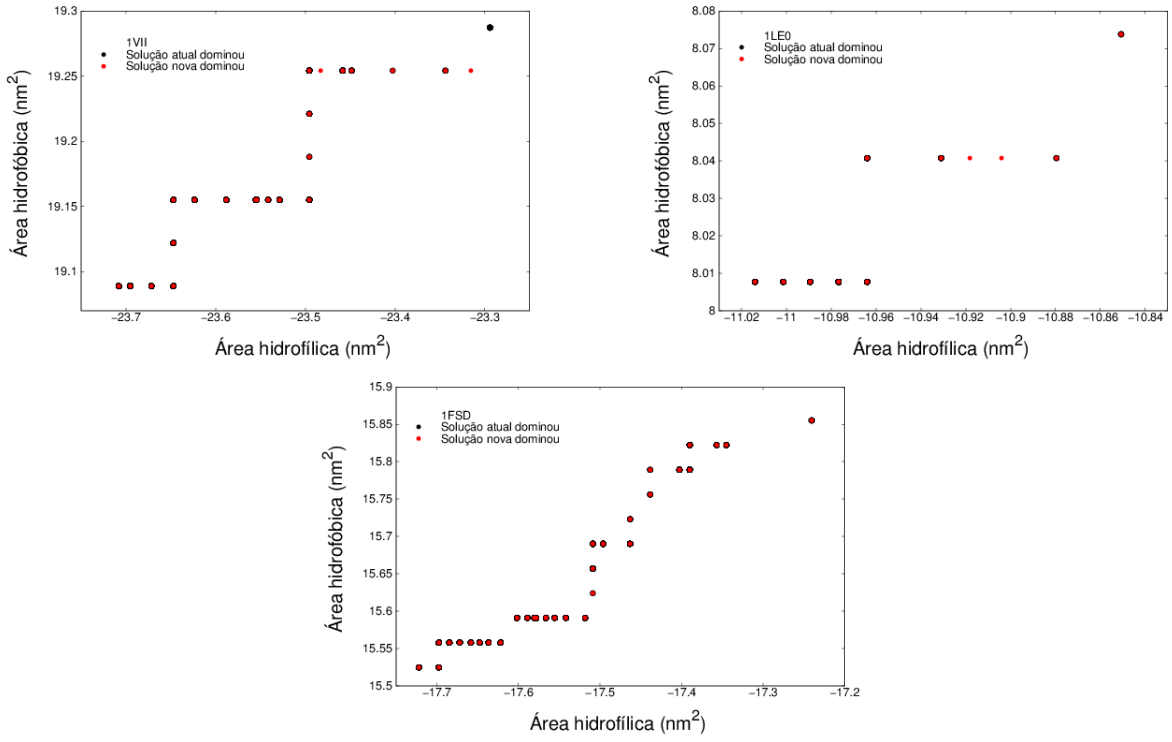
Fonte: autor.

Figura 58 – RMSD aplicando a função objetivo aSASA-pSASA no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

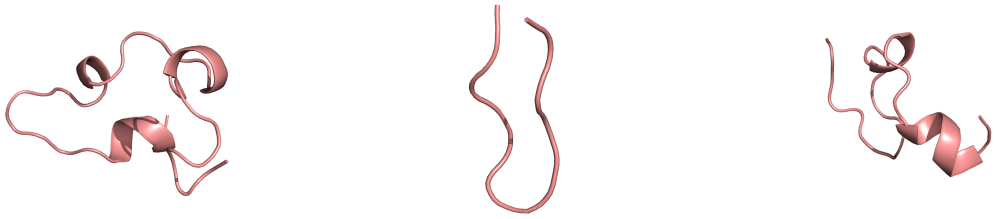
Figura 59 – Gráfico da área hidrofóbica (aSASA) em função da área hidrofílica (pSASA) no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

As Figuras 60 e 61 mostram as conformações estruturais após o processo de refinamento e o alinhamento com as proteínas-alvo, respectivamente.

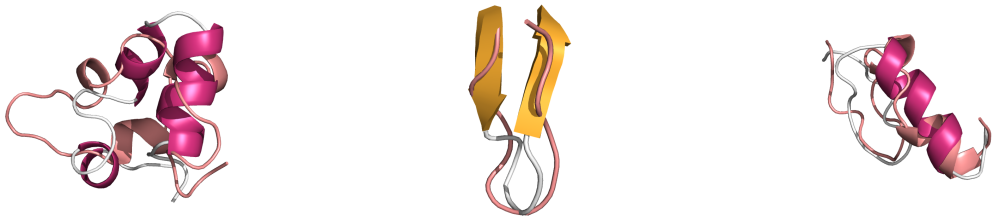
Figura 60 – Conformação estrutural refinada das proteínas preditas aplicando a função objetivo aSASA-pSASA via Monte Carlo com Dominância.



(a) 1VII PID 12. (b) 1LE0 PID 5. (c) 1FSD PID 8.

Fonte: autor.

Figura 61 – Alinhamento das estruturas preditas *versus* proteínas-alvo via Monte Carlo com Dominância.



(a) 1VII *vs.* 1VII PID 12. (b) 1LE0 *vs.* 1LE0 PID 5. (c) 1FSD *vs.* 1FSD PID 8.

Fonte: autor.

7.1.4.4 Energia Potencial e Energia de Solvatação

A Tabela 16 exibe os valores de RMSD calculados para as estruturas refinadas aplicando a função objetivo Potencial-GBSA:

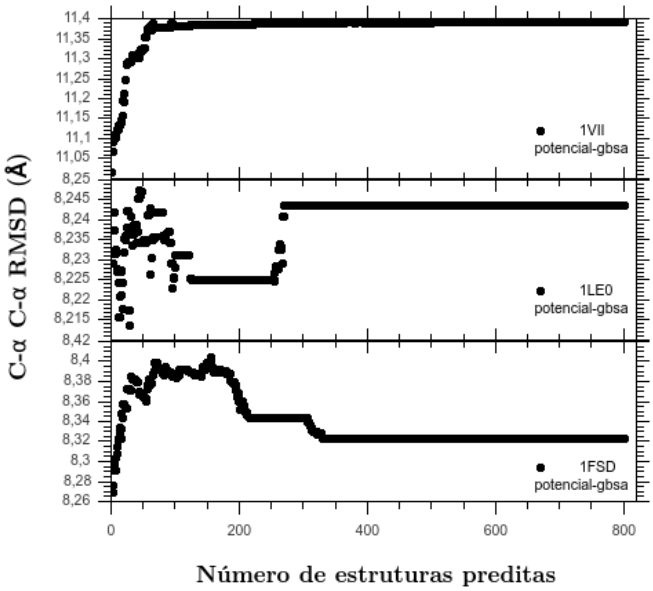
Tabela 16 – Valores de RMSD (mínimo) das estruturas refinadas (*ref*) aplicando a função objetivo Potencial-GBSA no algoritmo de Monte Carlo com Dominância.

Proteína-alvo	**RMSD**$^{ref}_{min}$ (Å)
1VII	7,349 (PID 1)
1LE0	3,409 (PID 28)
1FSD	4,063 (PID 2)

Fonte: autor.

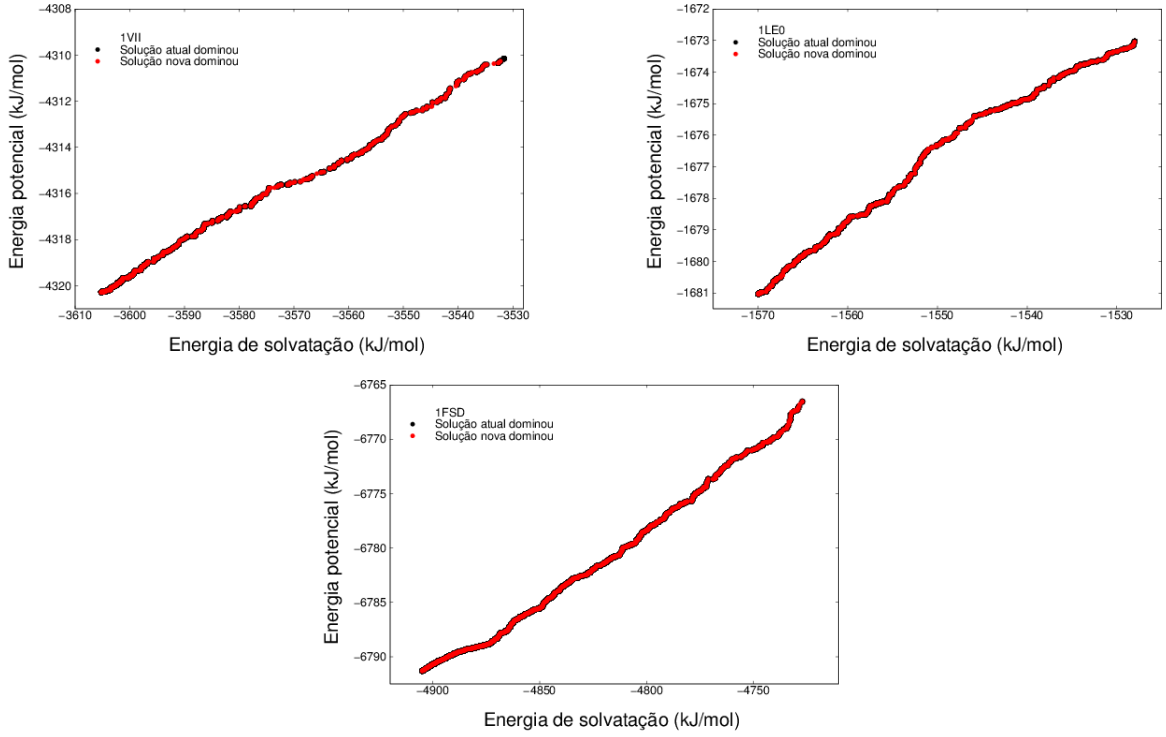
A Figura 62 exibe o comportamento do RMSD para as estrutura não refinadas, enquanto que o gráfico da Figura 63 mostra os valores da função objetivo Potencial-GBSA.

Figura 62 – RMSD aplicando a função objetivo Potencial-GBSA no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

Figura 63 – Gráfico da energia potencial em função da energia de solvatação no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

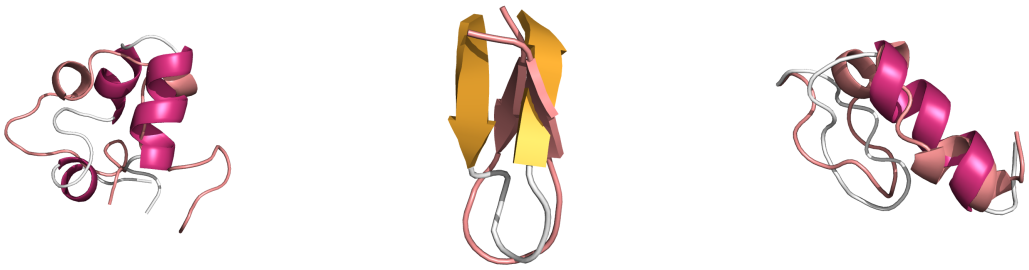
As Figuras 64 e 65 mostram as conformações estruturais após o processo de refinamento e o alinhamento com as proteínas-alvo, respectivamente.

Figura 64 – Conformação estrutural refinada das proteínas preditas aplicando a função objetivo Potencial-GBSA via Monte Carlo com Dominância.

(a) 1VII PID 1. (b) 1LE0 PID 28. (c) 1FSD PID 2.

Fonte: autor.

Figura 65 – Alinhamento das estruturas preditas *versus* proteínas-alvo via Monte Carlo com Dominância.



(a) 1VII *vs.* 1VII PID 1. (b) 1LE0 *vs.* 1LE0 PID 28. (c) 1FSD *vs.* 1FSD PID 2.

Fonte: autor.

7.1.4.5 Energia Potencial e Área Hidrofóbica

A Tabela 17 exibe os valores de RMSD calculados para as estruturas refinadas aplicando a função objetivo Potencial-aSASA:

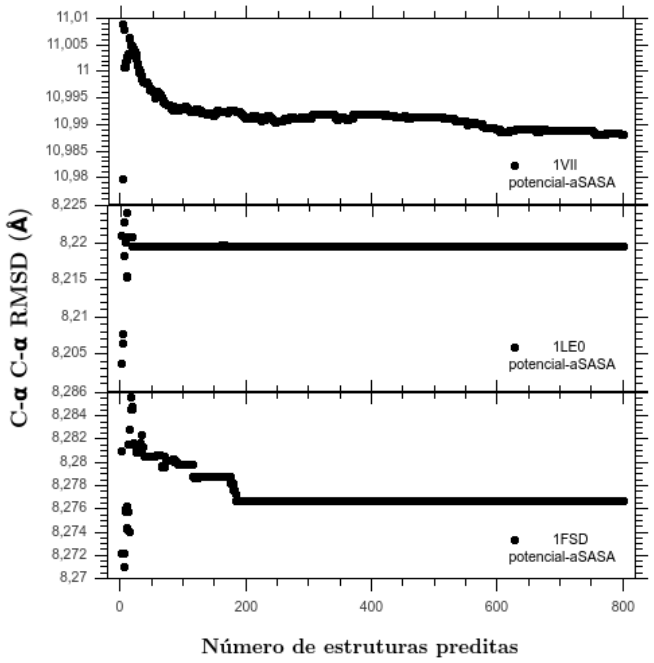
Tabela 17 – Valores de RMSD (mínimo) das estruturas refinadas (*ref*) aplicando a função objetivo Potencial-aSASA no algoritmo de Monte Carlo com Dominância.

Proteína-alvo	$\mathbf{RMSD}_{min}^{ref}$ (Å)
1VII	6,219 (PID 1)
1LE0	2,298 (PID 1)
1FSD	4,219 (PID 6)

Fonte: autor.

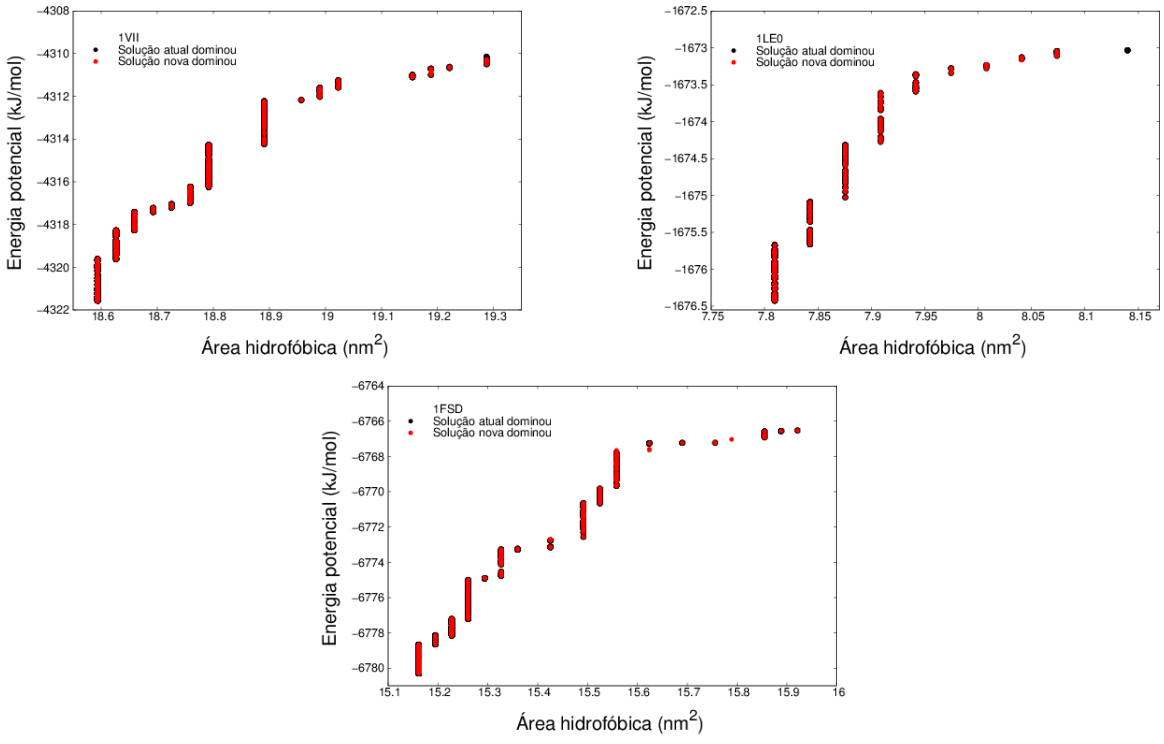
As Figuras 66 e 67 mostram o comportamento do RMSD para as estruturas não refinadas e os valores da função objetivo Potencial-aSASA, respectivamente.

Figura 66 – RMSD aplicando a função objetivo Potencial-aSASA no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

Figura 67 – Gráfico da energia potencial em função da área hidrofóbica no algoritmo de Monte Carlo com Dominância.



Fonte: autor.

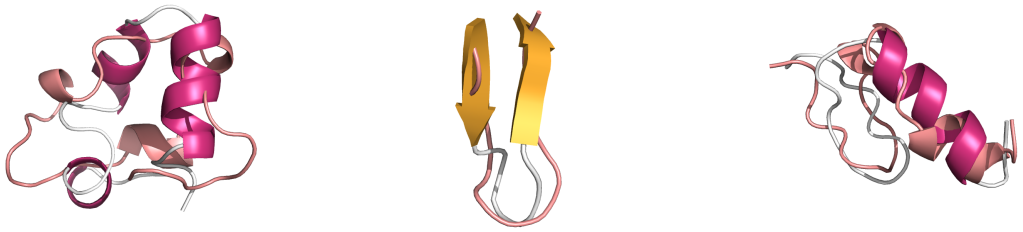
As Figuras 68 e 69 mostram as conformações estruturais após o processo de refinamento e o alinhamento com as proteínas-alvo, respectivamente.

Figura 68 – Conformação estrutural refinada das proteínas preditas aplicando a função objetivo Potencial-aSASA via Monte Carlo com Dominância.

(a) 1VII PID 1. (b) 1LE0 PID 1. (c) 1FSD PID 6.

Fonte: autor.

Figura 69 – Alinhamento das estruturas preditas *versus* proteínas-alvo via Monte Carlo com Dominância.



(a) 1VII *vs.* 1VII PID 1. (b) 1LE0 *vs.* 1LE0 PID 1. (c) 1FSD *vs.* 1FSD PID 6.

Fonte: autor.

7.2 Análise dos Resultados

Esta seção fará a análise dos resultados obtidos, confrontando-os com os objetivos que este trabalho pretendeu verificar (cap. 1, seção 1.4). Será interpretado tanto o comportamento dos RMSDs obtidos, quanto o comportamento de cada função objetivo e uma comparação de refinamento para cada uma das proteínas.

7.2.1 Custos Computacionais

Os experimentos *in silico* foram executados em um computador Dell Inspiron 13 Série 7000 Core i7 64 bits, 4 CPUs, 2,5GHz, 8GB de RAM, com HD híbrido de 500GB e 8GB de SSD. A Tabela 18 mostra os custos computacionais aproximados, em termos de tempo de CPU, gastos em cada execução das funções objetivos.

Tabela 18 – Custos computacionais gastos em termos de tempo de CPU, aproximadamente.

Objetivos	**1VII**	**1LE0**	**1FSD**
Potencial (Metropolis)	155 min	65 min	126 min
RG-GBSA	165 min	76 min	135 min
RG-pSASA	131 min	60 min	107 min
aSASA-pSASA	124 min	54 min	107 min
Potencial-GBSA	160 min	70 min	135 min
Potencial-pSASA	166 min	71 min	144 min

Fonte: autor.

7.2.2 Comportamento dos RMSDs

De acordo com as Tabelas 10 e 12 (estruturas sem refinamento), o método de Monte Carlo com Dominância, aplicando a função objetivo RG-GBSA, foi superior com RMSD menor para todas as três proteínas-alvo em comparação ao Método de Monte Carlo Metropolis e aos demais objetivos também. Convém ressaltar que todos os gráficos de RMSD foram considerados para estruturas não-refinadas a fim de analisar as estruturas preditas diretamente pelo 2PG.

No caso do gráfico de RMSD do Monte Carlo Metropolis (Figura 47), existem várias oscilações ao longo do número de estruturas salvas, aparentemente, não há nenhum tipo de padrão estrutural entre as proteínas preditas. No entanto, todos os gráficos de RMSD da Dominância (Figuras 50, 54, 58, 62 e 66) apresentam a tendência de convergir para um certo valor à medida que aumenta o número de passos de Monte Carlo. Uma possível explicação para esse fenômeno seja devido que a Dominância "restringe a variação de conformações", ou seja, é como se o caráter multiobjetivo reduzisse o intervalo de conformações que a proteína pudesse assumir, gerando estruturas semelhantes. Isto distoa dos resultados de Metropolis, onde pequenas variações dos parâmetros livres geram estruturas com valores de energia bem diferentes.

A Tabela 19 apresenta uma comparação dos melhores valores de RMSD das estruturas refinadas, reunindo as informações das Tabelas 11, 13, 14, 15, 16 e 17. Para a proteína 1VII, observa-se que todas as funções objetivos da dominância obtiveram resultados melhores que o de Metropolis. Entretanto, para a proteína 1LE0, Metropolis obteve resultados melhores. Para a 1FSD, Metropolis só ganhou em relação ao objetivo RG-GBSA, perdendo para todos os demais.

No caso das soluções obtidas por Dominância, desconsiderando Metropolis, a função objetivo RG-pSASA foi melhor para as proteínas 1VII e 1FSD, enquanto que para a proteína 1LE0 a Potencial-aSASA obteve um resultado ligeiramente melhor.

Tabela 19 – Valores de melhor RMSD (mínimo) das estruturas refinadas, comparando as predições entre Monte Carlo Metropolis e Monte Carlo com Dominância.

Objetivos	**1VII RMSD$^{ref}_{min}$** (Å)	**1LE0 RMSD$^{ref}_{min}$** (Å)	**1FSD RMSD$^{ref}_{min}$** (Å)
Potencial (Metropolis)	8,080 (PID 239)	2,066 (PID 536)	4,223 (PID 44)
RG-GBSA	6,932 (PID 223)	3,450 (PID 478)	4,451 (PID 714)
RG-pSASA	5,482 (PID 25)	2,306 (PID 42)	1,999 (PID 723)
aSASA-pSASA	5,691 (PID 12)	2,474 (PID 5)	3,988 (PID 8)
Potencial-GBSA	7,349 (PID 1)	3,409 (PID 28)	4,063 (PID 2)
Potencial-aSASA	6,219 (PID 1)	2,298 (PID 1)	4,219 (PID 6)

Fonte: autor.

Por conta da pequena diferença entre as funções objetivos RG-pSASA e Potencial-aSASA para a proteína-alvo 1LE0, pode-se considerar que a função objetivo RG-pSASA foi melhor para todos os casos dentre os objetivos da Dominância para as estruturas que passaram pelo algoritmo de refinamento.

A Tabela 20 mostra a variação percentual entre os RMSDs de Monte Carlo Metropolis e de Monte Carlo Dominância, dada pela equação:

$$\text{variação percentual} = \left[\frac{RMSD_{metropolis} - RMSD^{min}_{dominância}}{RMSD_{metropolis}}\right] \times 100 \qquad (33)$$

onde $RMSD_{metropolis}$ corresponde ao RMSD$^{ref}_{min}$ de Metropolis, enquanto que RMSD$^{min}_{dominância}$ corresponde ao **menor valor** de RMSD$^{ref}_{min}$ da Dominância.

Tabela 20 – Variação percentual entre os RMSDs de Monte Carlo Metropolis e de Monte Carlo com Dominância.

	Metropolis	**Dominância**		
Proteína	**RMSD$^{ref}_{min}$** (Å)	**RMSD$^{ref}_{min}$** (Å)	**Objetivo**	**Variação percentual**
1VII	8,080 (PID 239)	5,482 (PID 100)	RG-pSASA	≈ 32%
1LE0	2,066 (PID 536)	2,298 (PID 1)	Potencial-aSASA	≈ -11%
1FSD	4,223 (PID 44)	1,999 (PID 723)	RG-pSASA	≈ 53%

Fonte: autor.

A proteína-alvo 1FSD apresentou a maior diferença percentual (positiva), indicando uma variação de cerca de 53% da predição de Metropolis para a da Dominância. Segundo a Eq. (33), a diferença percentual positiva indica que sempre a predição por Dominância será melhor que a de Metropolis e, quanto maior for a diferença percentual positiva, mais a predição por Dominância prevalece sobre a de Metropolis. A mesma lógica se aplica para o caso contrário, quando a diferença percentual for negativa a predição por Metropolis prevalece sobre a da Dominância (caso da 1LE0).

A Tabela 21 faz uma comparação entre os RMSDs das últimas conformações refinadas de cada proteína-alvo:

Tabela 21 – Valores de RMSD das últimas estruturas refinadas (PID 800), comparando as predições entre Monte Carlo Metropolis e Monte Carlo com Dominância.

Objetivos	**1VII RMSD**$^{ref}_{\text{PID 800}}$ (Å)	**1LE0 RMSD**$^{ref}_{\text{PID 800}}$ (Å)	**1FSD RMSD**$^{ref}_{\text{PID 800}}$ (Å)
Potencial (Metropolis)	8,296	7,906	8,838
RG-GBSA	7,270	4,634	6,596
RG-pSASA	7,528	8,198	8,324
aSASA-pSASA	7,365	8,269	8,381
Potencial-GBSA	7,493	8,220	8,511
Potencial-aSASA	5,847	8,263	8,350

Fonte: autor.

O RMSD é um meio de se medir a similaridade entre duas estruturas. Para duas estruturas idênticas, o RMSD é nulo. Apesar dos valores de RMSD obtidos neste trabalho estarem ainda altos, tanto para Metropolis quanto para a Dominância, de acordo com a Figura 33, isto significa que a estrutura predita ainda se encontra em um caminho intermediário do funil de energia livre. Em razão disto é que as estruturas preditas apresentam diferenças estruturais como mostram as figuras de alinhamento.

7.2.3 Comportamento das Funções Objetivos

Porém, a despeito dessas diferenças, o método de Monte Carlo com Dominância, aplicando a função objetivo RG-GBSA para as proteínas não-refinadas, obteve melhor resposta em relação ao largamente empregado método de Monte Carlo Metropolis para a predição de estruturas terciárias de proteínas. Após o refinamento, verificou-se que em dois casos (1VII e 1FSD), a função objetivo RG-pSASA obteve melhor resposta que o critério de Metropolis. O tratamento multiobjetivo, com excessão do caso da proteína 1LE0, permitiu uma melhor exploraração do espaço de busca ao considerar mais de um objetivo para a solução do problema, confirmando a hipótese inicial deste trabalho.

Para as simulações deste trabalho, considerando as proteínas-alvo 1VII, 1LE0 e 1FSD, os objetivos RG, GBSA e pSASA destacaram-se dos demais analisados. Enquanto que a energia potencial de Monte Carlo Metropolis depende exclusivamente dos parâmetros do campo de força utilizado pela Dinâmica Molecular, os objetivos RG, GBSA e pSASA consideram apenas os aspectos estruturais da proteína. Isto permitiu concluir nos testes realizados que a busca pelo estado nativo por Dominância obteve melhor resposta quando se mede o modo como a proteína se empacota ao longo do tempo (RG), o solvente em que ela se encontra (GBSA) e a área superficial polar acessível ao solvente (pSASA).

É interessante notar o comportamento das funções objetivos nos gráficos das Figuras 51, 55, 59, 63 e 67. Observa-se uma relação praticamente linear entre os objetivos energéticos Potencial-GBSA (Figura 63), enquanto que o gráfico de RG-GBSA (Figura 51) assemelha-se muito ao de uma exponencial. Com exceção do raio de giro, todas as outras funções objetivos envolvendo propriedades estruturais apresentam um gráfico com aparência de uma função Heaviside (função degrau).

Não existe uma modelagem analítica para a predição de estruturas de proteínas, por isso que se utiliza técnicas de otimização para a solução do PSP. Os padrões de Dominância observados neste trabalho (linear, exponencial e Heaviside) incitam uma investigação mais detalhada que foge do escopo deste trabalho de mestrado, porém são evidências que merecem um estudo posterior.

CAPÍTULO 8

Conclusões

Neste trabalho foi desenvolvido o Monte Carlo com Dominância, um novo método de predição de estruturas terciárias de proteínas empregando algoritmos genéticos como técnica de otimização multiobjetivo no algoritmo de Monte Carlo. Foi realizada uma comparação entre o critério de Metropolis, estritamente energético, com o critério de Dominância, que envolve tanto as propriedades energéticas quanto estruturais da proteína. Para o cálculo das propriedades físicas da proteína foi utilizado o *software* GROMACS, enquanto que para a otimização multiobjetivo utilizou-se o *framework* ProtPred-Gromacs (2PG).

No que diz respeito à abordagem multiobjetivo foram considerados os seguintes objetivos: energia potencial, energia de solvatação, raio de giro, área hidrofílica e área hidrofóbica. Para a predição das estruturas foram selecionadas três proteínas-alvo: 1VII, 1LE0 e 1FSD. A fim de melhorar a qualidade da predição, as estruturas preditas passaram por um algoritmo de refinamento estrutural de minimização energética a nível atômico denominado ModRefiner. A forma de mensurar a qualidade da predição em relação à estrutura nativa foi feita pelo cálculo de RMSD entre as estruturas.

O objetivo da análise multiobjetivo na escolha de uma estrutura no espaço de busca é que esta possui mais de um critério (objetivo) para a tomada de decisão. Com isto, se espera que a predição seja melhorada considerando outros objetivos. Com base nos resultados obtidos, foi possível verificar que para duas proteínas, a 1VII e a 1FSD, a abordagem multiobjetivo do Monte Carlo com Dominância produziu um efeito significativo, em especial para a 1FSD aplicando a função objetivo RG-pSASA, que chegou a ser de 53% a diferença em comparação ao tradicional método de Monte Carlo Metropolis.

Portanto, conclui-se que o método de Monte Carlo com Dominância obteve um êxito considerável para duas das três proteínas-alvo analisadas neste trabalho. Destaca-se, em especial, o comportamento dos RMSDs e dos gráficos das funções objetivos, podendo haver indícios de que o padrão observado possa se repetir para várias outras proteínas, configurando uma característica peculiar do sistema. De certo modo, tal resultado não

estava previsto e possivelmente fornece informações inéditas a respeito da predição de estruturas terciárias de proteínas.

8.1 Trabalhos Futuros

Para trabalhos futuros pretende-se melhorar a qualidade dos RMSDs objetivando aproximar cada vez mais da estrutura nativa, ou estrutura-alvo, como por exemplo inserindo novos objetivos ou mesmo novas combinações na função objetivo. Também é de interesse executar os mesmos experimentos considerando proteínas com tamanhos maiores do que aquelas analisadas neste trabalho, possibilitando investigar se os padrões observados nos gráficos se conservam. Se isto ocorrer, então pode existir alguma relação entre os objetivos e os padrões observados, como por exemplo o caráter linear e as semelhanças com funções Heaviside para algumas das soluções não-dominadas, de modo que se possa inferir se esses comportamentos estão sempre correlacionados ou não.

Novas abordagens poderão ser feitas também, como elaborar um novo tipo de rotação dos ângulos diedros. Atualmente o 2PG rotaciona a conformação em 4 tipos: ϕ, ψ, ω e χ. Contudo, tais rotações não são tão eficientes quando a conformação já está compactada, pois existe a possibilidade de que mais choques entre átomos possam ocorrer e, assim, acarretando em energias mais altas. Logo, essa nova conformação será rejeitada pelos critérios de mínimos de energia. Uma sugestão é elaborar um novo tipo de rotação que ocorra em duas direções: i) no sentido do centro de massa da proteína, quando o resíduo for hidrofóbico; e ii) no sentido contrário do centro de massa da proteína, quando o resíduo for hidrofílico.

Referências Bibliográficas

ANFINSEN, C. B. Principles that govern the folding of protein chains. **Science**, v. 181, p. 223–230, 1973. Citado 2 vezes nas páginas 20 e 43.

ASTUTI, A. D.; MUTIARA, A. B. Performance Analysis on Molecular Dynamics Simulation of Protein Using GROMACS. **CoRR**, abs/0912.0893, 2009. Disponível em: <http://arxiv.org/abs/0912.0893>. Citado na página 81.

BAKER, E.; HUBBARD, R. Hydrogen bonding in globular proteins. **Prog. Biophy. Mol. Biol.**, v. 44, p. 97–179, 1984. Citado na página 40.

BARTON, G.; COHEN, P.; BRADFORD, D. Conservation analysis and structure prediction of the protein serine/threonine phosphatases. **Eur. J. Biochem**, v. 220, p. 225–237, 1993. Citado na página 55.

BHATTI, M. A. **Practical Optimization Methods: With Mathematica® Applications**. 2000. ed. New York: Springer, 2000. 715 p. ISBN 0387986316. Citado na página 23.

BIOBLENDER. 2015. Disponível em: <http://www.bioblender.eu>. Acesso em: 14 maio 2015. Citado na página 53.

BLAST, N. 2015. Disponível em: <http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml>. Acesso em: 30 maio 2015. Citado na página 51.

BRADLEY, P.; MISURA, K.; BAKER, D. Toward high-resolution *de novo* structure prediction for small proteins. **Science**, v. 309, n. 5742, p. 1868–1871, 2005. Citado na página 55.

BRANDEN, C.; TOOZE, J. **Introduction to Protein Structure**. [S.l.]: Garland Publishing, 1991. Citado na página 48.

BROOKS, B. R. et al. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. **Journal of Computational Chemistry**, John Wiley & Sons, Inc., v. 4, n. 2, p. 187–217, 1983. ISSN 1096-987X. Disponível em: <http://dx.doi.org/10.1002/jcc.540040211>. Citado 2 vezes nas páginas 56 e 57.

CARTER, C. W.; WOLFENDEN, R. tRNA acceptor stem and anticodon bases form independent codes related to protein folding. **Proceedings of the National Academy of Sciences**, 2015. Disponível em: <http://www.pnas.org/content/early/2015/05/27/1507569112.abstract>. Citado na página 20.

CASP. 2015. Disponível em: <http://predictioncenter.org>. Acesso em: 12 jun. 2015. Citado na página 49.

CHARMM. 2015. Disponível em: <http://www.charmm.org/>. Acesso em: 02 maio 2015. Citado na página 60.

CHENG, Y. et al. A Primer to Single-Particle Cryo-Electron Microscopy. **Cell**, v. 161, n. 3, p. 438 – 449, Apr 2015. ISSN 0092-8674. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0092867415003700>. Citado na página 22.

CHOU, P.; FASMAN, G. Empirical predictions of protein conformation. **Annu. Rev. Biochem.**, v. 47, p. 251–76, 1978. Citado na página 57.

COCHRAN, A. G.; SKELTON, N. J.; STAROVASNIK, M. A. Tryptophan zippers: stable, monomeric beta -hairpins. **Proc. Natl. Acad. Sci. U.S.A.**, v. 98, n. 10, p. 5578–5583, 2001. Citado na página 96.

COELLO, C. A. C. Evolutionary multi-objective optimization: a historical view of the field. **Computational Intelligence Magazine, IEEE**, IEEE, v. 1, n. 1, p. 28–36, Feb 2006. Citado na página 61.

COHEN, B.; PRESNELL, S.; COHEN, F. Origins of structural diversity within sequentially identical hexapeptides. **Protein Science**, v. 2, p. 2134–2145, 1993. Citado na página 54.

COHON, J. **Multiobjective Programing and Plannig**. New York: Academic Press, 1978. 352 p. ISBN 0486432637. Citado na página 65.

CREIGHTON, T. **Proteins: Structures and Molecular Properties**. 2. ed. New York: W.H. Freeman & Co, 1993. Citado na página 39.

CSSB SYSTEMS BIOLOGY. 2015. Disponível em: <http://cssb.biology.gatech.edu/skolnick/webservice/MetaTASSER>. Acesso em: 25 jun. 2015. Citado na página 56.

CUI, Y.; CHEN, R.; WONG, W. Protein Folding Simulation With Genetic Algorithm and SuperSecondary Structure Constraints. **Proteins**, v. 31, p. 247–257, 1998. Citado na página 67.

CUTELLO, V.; NARZISI, G.; NICOSIA, G. A multi-objective evolutionary approach to the protein structure prediction problem. **J. R. Soc. Interface**, v. 83, p. 1–13, 2005. Citado na página 67.

DAHIYAT, B. I.; MAYO, S. L. De novo protein design: fully automated sequence selection. **Science**, v. 278, n. 5335, p. 82–87, 1997. Citado na página 96.

DARWIN, C. **On the Origin of Species By Means of Natural Selection**. [S.l.]: Gramercy, 1859. Citado 3 vezes nas páginas 20, 67 e 69.

DEB, K. Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems. **Evolutionary Computation**, v. 7, p. 205–230, 1998. Citado na página 62.

_____. **Multi-Objective Optimization using Evolutionary Algorithms**. [S.l.]: John Wiley and Sons, 2001. ISBN 047187339X. Citado 4 vezes nas páginas 23, 62, 63 e 69.

DEB, K. et al. **A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II**. KanGAL report number 200001. Indian Institute of Technology, Kanpur, India, 2000. Citado na página 67.

DEB, K.; MOHAN, M.; MISHRA, S. **A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions**. KanGAL report number 2003002. Indian Institute of Technology, Kanpur, India, 2003. Citado na página 63.

DEJONG, K. A. **Evolutionary Computation**. [S.l.]: The MIT Press, 2006. ISBN 0262041944. Citado 2 vezes nas páginas 23 e 69.

DILL, K. A.; BROMBERG, S. **Molecular Driving Forces: Statistical Thermodynamics in Chemistry & Biology**. 1. ed. [S.l.]: Garland Science, 2002. ISBN 9780815320517. Citado 3 vezes nas páginas 45, 46 e 47.

DILL, K. A. et al. The Protein Folding Problem. **Annual Review of Biophysics**, v. 37, n. 1, p. 289–316, 2008. Citado 4 vezes nas páginas 21, 22, 43 e 44.

DRENTH, J. **Principles of Protein X-ray Crystallography**. [S.l.]: Springer, 1994. 368 p. Citado 2 vezes nas páginas 22 e 47.

DUAN, Y.; KOLLMAN, P. A. Pathways to a Protein Folding Intermediate Observed in a 1-Microsecond Simulation in Aqueous Solution. **Science**, v. 282, n. 5389, p. 740–744, 1998. Disponível em: <http://www.sciencemag.org/content/282/5389/740.abstract>. Citado na página 59.

EARL, D. J.; DEEM, M. W. Monte Carlo Simulations. In: KUKOL, A. (Ed.). **Molecular Modeling of Proteins**. Humana Press, 2008, (Methods Molecular Biology™, v. 443). p. 25–36. ISBN 978-1-58829-864-5. Disponível em: <http://dx.doi.org/10.1007/978-1-59745-177-2_2>. Citado 2 vezes nas páginas 84 e 87.

ECHENIQUE, P. Introduction to protein folding for physicists. **Contemporary Physics**, v. 48, n. 2, p. 81–108, 2007. Citado 3 vezes nas páginas 22, 54 e 60.

FACCIOLI, R. A. **Implementação de um Framework de Computação Evolutiva Multi-Objetivo para Predição *Ab Initio* da Estrutura Terciária de Proteínas**. Tese (Doutorado em Engenharia Elétrica) — Escola de Engenharia de São Carlos - Universidade de São Paulo, São Carlos, 2012. Citado 6 vezes nas páginas 23, 36, 43, 45, 69 e 92.

FACCIOLI, R. A. ***Framework* Evolutivo Multi-objetivo para Predição *Ab Initio* de Estruturas de Proteínas**. Instituto de Ciências Matemáticas e de Computação – Departamento de Ciências de Computação, Universidade de São Paulo, São Paulo, 2015. Citado na página 71.

_____. **2PG Github Repositório**. 2016. Disponível em: <https://github.com/rodrigofaccioli/2pg_cartesian>. Acesso em: 06 maio 2016. Citado na página 70.

FACCIOLI, R. A. et al. Protpred-gromacs: Evolutionay algorithm with gromacs for protein structure prediction. **BIOMAT 2011 International Symposium on Mathematical and Computational Biology**, p. 1–12, 2011. Citado na página 70.

FISCHER, D. Servers for protein structure prediction. **Curr. Opin. Struct. Biol.**, v. 16, n. 2, p. 178–182, Apr 2006. Citado na página 60.

FLIEGE, J.; DRUMMOND, L. M. G. n.; SVAITER, B. F. Newton's Method for Multiobjective Optimization. **SIAM J. on Optimization**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, v. 20, n. 2, p. 602–626, May 2009. ISSN 1052-6234. Disponível em: <http://dx.doi.org/10.1137/08071692X>. Citado na página 65.

FLIEGE, J.; SVAITER, B. F. Steepest descent methods for multicriteria optimization. **Mathematical Methods of Operations Research**, Springer-Verlag Berlin Heidelberg, v. 51, n. 3, p. 479–494, 2000. ISSN 1432-2994. Disponível em: <http://dx.doi.org/10.1007/s001860000043>. Citado na página 65.

FOGEL, D. B. An introduction to simulated evolutionary computation. **IEEE Transactions on Neural Networks**, v. 5, p. 3–14, 1994. Citado na página 67.

FOGEL, L.; OWENS, A.; WALSH, M. **Artificial Intelligence through Simulated Evolution**. [S.l.]: John Wiley, 1966. Citado na página 67.

FONSECA, C.; FLEMING, P. An overview of evolutionary algorithms in multiobjective optimization. **Evolutionary Computation**, v. 3, n. 1, p. 1–16, 1995. Citado 2 vezes nas páginas 62 e 64.

FORCEFIELD BASED SIMULATIONS. 2015. Disponível em: <http://www.chem.cmu.edu/courses/09-560/docs/msi/ffbsim/B_AtomTypes.html#639162>. Acesso em: 25 jun. 2015. Citado na página 56.

GINALSKI, K. Comparative modeling for protein structure prediction. **Current Opinion in Structural Biology**, v. 16, n. 2, p. 172–177, 2006. ISSN 0959-440X. Citado na página 48.

GLOVER, F.; LAGUNA, M. **Tabu Search**. [S.l.]: Springer, 1997. 382 p. Citado na página 66.

GOLDBERG, D. E. **Genetic Algorithms in Search, Optimization, and Machine Learning**. Reading, Massachusetts: Addison-Wesley Publishing Company, 1989. Citado na página 67.

GROMACS. 2015. Disponível em: <http://www.gromacs.org/About_Gromacs>. Acesso em: 14 maio 2015. Citado 4 vezes nas páginas 23, 75, 79 e 80.

_____. 2015. Disponível em: <http://www.gromacs.org/Documentation/File_Formats>. Acesso em: 14 maio 2015. Citado na página 78.

GUNSTEREN, W. F. van et al. **Biomolecular Simulation: The GROMOS96 manual and userguide**. Zürich, Switzerland: Hochschuleverlag AG an der ETH Zürich, 1996. Disponível em: <http://amzn.com/3728124222>. Citado na página 57.

HAGLER, A. T.; HULER, E.; LIFSON, S. Energy functions for peptides and proteins. I. Derivation of a consistent force field including the hydrogen bond from amide crystals. **Journal of the American Chemical Society**, v. 96, n. 17, p. 5319–5327, 1974. Disponível em: <http://dx.doi.org/10.1021/ja00824a004>. Citado na página 57.

HAIMES, Y. Y.; LASDON, L. S.; WISMER, D. A. On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization. **IEEE Transactions on Systems, Man, and Cybernetics**, v. 1, p. 296–297, 1971. Citado na página 65.

HASTINGS, W. K. Monte Carlo sampling methods using Markov chains and their applications. **Biometrika**, v. 57, n. 1, p. 97–109, 1970. Disponível em: <http://biomet.oxfordjournals.org/content/57/1/97.abstract>. Citado na página 86.

HESS, B. et al. GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. **Journal of Chemical Theory and Computation**, Stockholm Center for Biomembrane Research, Stockholm University, SE-10691 Stockholm, Sweden, v. 4, n. 3, p. 435–447, Mar 2008. Citado na página 23.

HILBERT, M.; BÖHM, G.; JAENICKE, R. Structural relationships of homologous proteins as a fundamental principle in homology modeling. **Proteins**, v. 17, p. 138–151, 1993. Citado 2 vezes nas páginas 48 e 54.

HOLLAND, J. **Adaptation in natural and artificial systems**. [S.l.]: University of Michigan Press, 1975. Citado na página 67.

HORN, J. **Handbook of Evolutionary Computation**. Oxford, England: Oxford University Press, 1997. Citado na página 64.

ISHIBUCHI, H.; TSUKAMOTO, N.; NOJIMA, Y. Evolutionary many-objective optimization: A short review. In: **2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)**. [S.l.: s.n.], 2008. p. 2419–2426. ISSN 1089-778X. Citado na página 68.

JAIMES, A. L.; COELLO, C. A. C. An Introduction to Multi-Objective Evolutionary Algorithms and some of Their Potential Uses in Biology. In: SMOLINSKI, T.; MILANOVA, M. G.; HASSANIEN, A.-E. (Ed.). **Applications of Computational Intelligence in Biology: Current Trends and Open Problems**. Berlin: Springer, 2008. p. 79–102. Citado na página 91.

JAUCH, R. et al. Assessment of CASP7 structure predictions for template free targets. **Proteins**, v. 69, n. Suppl. 8, p. 57–67, 2007. Citado na página 55.

JMOL. 2015. Disponível em: <http://jmol.sourceforge.net>. Acesso em: 07 maio 2015. Citado 4 vezes nas páginas 12, 52, 53 e 60.

JONES, D.; TAYLOR, W.; THORNTON, J. A new approach to protein fold recognition. **Nature**, v. 358, p. 86–89, 1992. Citado na página 55.

JORGENSEN, W. L.; TIRADO-RIVES, J. The OPLS [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin. **Journal of the American Chemical Society**, v. 110, n. 6, p. 1657–1666, 1988. Disponível em: <http://dx.doi.org/10.1021/ja00214a001>. Citado na página 57.

KABSCH, W.; SANDER, C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. **Biopolymers**, v. 22, p. 2577–2637, 1983. Citado 2 vezes nas páginas 39 e 54.

KACZANOWSKI, S.; ZIELENKIEWICZ, P. Why similar protein sequences encode similar three-dimensional structures? **Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)**, v. 125, n. 3, p. 643–650, Mar. 2010. ISSN 1432-881X. Citado na página 55.

KENDREW, J. C. et al. A three-dimensional model of the myoglobin molecule obtained by x-ray analysis. **Nature**, v. 181, p. 662–666, 1958. Citado na página 27.

KHOKHLOV, A. R.; GROSBERG, A. Y.; PANDE, V. S. **Statistical Physics of Macromolecules**. 1. ed. New York: AIP-Press, 1994. 350 p. ISBN 978-1-56396-071-0. Citado na página 87.

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by Simulated Annealing. **Science**, v. 220, n. 4598, p. 671–680, 1983. Disponível em: <http://www.sciencemag.org/content/220/4598/671.abstract>. Citado na página 58.

KLEPEIS, J. L.; FLOUDAS, C. A. ASTRO-FOLD: A Combinatorial and Global Optimization Framework for Ab Initio Prediction of Three-Dimensional Structures of Proteins from the Amino Acid Sequence. **Biophysical Journal**, Elsevier, v. 85, n. 4, p. 2119–2146, Jun 2003. Disponível em: <http://dx.doi.org/10.1016/S0006-3495(03)74640-2>. Citado 2 vezes nas páginas 56 e 59.

KOSLOVER, E. F.; WALES, D. J. Geometry optimization for peptides and proteins: comparison of Cartesian and internal coordinates. **The Journal of chemical physics**, v. 127, n. 23, p. 234105, 2007. Citado na página 50.

LAZARIDIS, T.; KARPLUS, M. Discrimination of the native from misfolded protein models with an energy function including implicit solvation. **J. Mol. Biol.**, v. 288, n. 3, p. 477–487, May 1999. Citado na página 60.

LEE, J.; SCHERAGA, H. A.; RACKOVSKY, S. Conformational analysis of the 20-residue membrane-bound portion of melittin by conformational space annealing. **Biopolymers**, v. 46, n. 2, p. 103–116, Aug 1998. Citado na página 59.

LEE, J.; WU, S.; ZHANG, Y. ***Ab Initio* Protein Structure Prediction**. 2015. Disponível em: <http://zhanglab.ccmb.med.umich.edu/papers/2009_4.pdf>. Acesso em: 12 jun. 2015. Citado 4 vezes nas páginas 23, 48, 56 e 59.

LESSEL, U.; SCHOMBURG, D. Similarities between protein 3-D structures. **Protein Eng.**, v. 10, n. 7, p. 1175–87, Oct 1994. Citado na página 54.

LEVINTHAL, C. Are there pathways for protein folding? **Journal de Chimie Physique et de Physico-Chimie Biologique**, v. 65, p. 44–45, 1968. Citado na página 43.

LI, H.; TANG, C.; WINGREEN, N. S. Nature of Driving Force for Protein Folding: A Result From Analyzing the Statistical Potential. **Phys. Rev. Lett.**, American Physical Society, v. 79, p. 765–768, Jul 1997. Citado 3 vezes nas páginas 46, 49 e 94.

LIMA, T. et al. Multi-objective evolutionary approach to ab initio protein tertiary structure prediction. **BIOMAT 2006 International Symposium on Mathematical and Computational Biology,**, p. 269–286, 2006. Citado na página 70.

LINDAHL, E.; HESS, B.; SPOEL, D. van der. GROMACS 3.0: A package for molecular simulation and trajectory analysis. **Journal of Molecular Modeling**, v. 7, p. 306–317, 2001. Citado na página 76.

LINDORFF-LARSEN, K. et al. Improved side-chain torsion potentials for the Amber ff99SB protein force field. **Proteins**, v. 78, n. 8, p. 1950–1958, Jun 2010. Citado 2 vezes nas páginas 75 e 92.

LODISH, H. et al. **Biologia Celular e Molecular**. [S.l.]: Artmed, 2004. Citado na página 42.

LOTAN, I.; SCHWARZER, F.; LATOMBE, J.-C. Efficient Energy Computation for Monte Carlo Simulation of Proteins. In: BENSON, G.; PAGE, R. (Ed.). **Algorithms in Bioinformatics**. Springer Berlin Heidelberg, 2003, (Lecture Notes in Computer Science, v. 2812). p. 354–373. ISBN 978-3-540-20076-5. Disponível em: <http://dx.doi.org/10.1007/978-3-540-39763-2_26>. Citado na página 86.

LUND, M.; TRULSSON, M.; PERSSON, B. Faunus: An object oriented framework for molecular simulation. **Source code for biology and medicine**, v. 3, n. 1, Feb 2008. ISSN 1751-0473. Citado na página 23.

LUTHY, R.; BOWIE, J. U.; EISENBERG, D. Assessment of protein models with three-dimensional profiles. **Nature**, v. 356, n. 6364, p. 83–85, Mar 1992. Citado na página 60.

MACKERELL, A. D. et al. All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins. **The Journal of Physical Chemistry B**, v. 102, n. 18, p. 3586–3616, 1998. Disponível em: <http://dx.doi.org/10.1021/jp973084f>. Citado na página 92.

MAIOROV, V. N.; CRIPPEN, G. M. Significance of root-mean-square deviation in comparing three-dimensional structures of globular proteins. **Journal of Molecular Biology**, v. 235, n. 2, p. 625 – 634, 1994. ISSN 0022-2836. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0022283684710175>. Citado na página 25.

MCKNIGHT, C. J.; MATSUDAIRA, P. T.; KIM, P. S. NMR structure of the 35-residue villin headpiece subdomain. **Nat. Struct. Biol.**, v. 4, n. 3, p. 180–184, 1997. Citado na página 96.

METROPOLIS, N. et al. Equation of State Calculations by Fast Computing Machines. **J. Chem. Phys.**, v. 21, p. 1087–1092, Jun 1953. Disponível em: <http://adsabs.harvard.edu/abs/1953JChPh..21.1087M>. Citado na página 85.

METROPOLIS, N.; ULAM, S. The Monte Carlo method. **J. Am. Stat. Assoc.**, v. 44, n. 247, p. 335–341, Sep 1949. Citado na página 82.

MICHALEWICZ, Z.; SCHOENAUER, M. Evolutionary Algorithms for Constrained Parameter Optimization Problems. **Evolutionary Computation**, v. 4, p. 1–32, 1996. Citado 2 vezes nas páginas 67 e 69.

MODREFINER. 2016. Disponível em: <http://zhanglab.ccmb.med.umich.edu/ModRefiner>. Acesso em: 19 abr. 2016. Citado na página 98.

MOHAN, K. S. et al. CADB-2.0: Conformation Angles Database. **Acta Crystallographica Section D**, v. 61, n. 5, p. 637–639, May 2005. Citado na página 68.

NAMBA, A. M.; SILVA, V. B. d.; SILVA, C. H. T. P. d. Dinâmica molecular: teoria e aplicações em planejamento de fármacos. **Eclética Química**, Scielo, v. 33, p. 13 – 24, Dec 2008. ISSN 0100-4670. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-46702008000400002&nrm=iso>. Citado na página 92.

NCBI - NATIONAL CENTER FOR BIOTECHNOLOGY INFORMATION. 2015. Disponível em: <http://www.ncbi.nlm.nih.gov>. Acesso em: 02 jun. 2015. Citado na página 52.

NELSON, D. L.; COX, M. M. **Lehninger: Principles of Biochemistry**. 5. ed. New York: W.H. Freeman and Company, 2008. ISBN 071677108X. Citado 5 vezes nas páginas 20, 21, 27, 34 e 45.

NOBELPRIZE.ORG. 2015. Disponível em: <http://www.nobelprize.org/nobel_prizes/chemistry/laureates/1972/press.html>. Acesso em: 12 jun. 2015. Citado na página 43.

OłDZIEJ, S. et al. Physics-based protein-structure prediction using a hierarchical protocol based on the UNRES force field: Assessment in two blind tests. **Proceedings of the National Academy of Sciences of the United States of America**, v. 102, n. 21, p. 7547–7552, 2005. Disponível em: <http://www.pnas.org/content/102/21/7547.abstract>. Citado 2 vezes nas páginas 55 e 56.

PANDE, V. S.; ROKHSAR, D. S. Folding pathway of a lattice model for proteins. **Proceedings of the National Academy of Sciences**, v. 96, n. 4, p. 1273–1278, Feb 1999. Citado na página 43.

PARSONS, J. et al. Practical conversion from torsion space to Cartesian space for in silico protein synthesis. **J. Comput. Chem.**, Wiley Subscription Services, Inc., A Wiley Company, Texas Agricultural Experiment Station, Texas A&M University, College Station, Texas 77843, USA., v. 26, n. 10, p. 1063–1068, 2005. Citado 2 vezes nas páginas 72 e 76.

PONDER, J. **Tinker Software Tools for Molecular Design. Washington University, Saint Louis**. 2001. Citado na página 23.

PYMOL. 2015. Disponível em: <https://www.pymol.org>. Acesso em: 14 maio 2015. Citado na página 52.

RAMACHANDRAN, G.; SASISKHARAN, V. Conformation of polypeptides and proteins. **Protein Chem.**, v. 23, p. 283–437, 1968. Citado 2 vezes nas páginas 36 e 49.

RAMACHANDRAN PLOT. 2015. Disponível em: <http://www.cryst.bbk.ac.uk/PPS95/course/3_geometry/rama.html>. Acesso em: 28 jun. 2015. Citado na página 37.

RASMOL. 2015. Disponível em: <http://rasmol.org>. Acesso em: 14 maio 2015. Citado na página 52.

RCSB PROTEIN DATA BANK. 2015. Disponível em: <http://www.rcsb.org/pdb/statistics/holdings.do>. Acesso em: 25 jun. 2015. Citado na página 21.

_____. 2015. Disponível em: <http://www.rcsb.org/pdb/explore/jmol.do?structureId=1MBN&bionumber=1>. Acesso em: 23 abr. 2015. Citado 2 vezes nas páginas 27 e 53.

_____. 2015. Disponível em: <http://www.rcsb.org/pdb/explore/explore.do?structureId=4TNC>. Acesso em: 25 maio 2015. Citado 2 vezes nas páginas 41 e 42.

_____. 2015. Disponível em: <http://www.rcsb.org/pdb/home/home.do>. Acesso em: 14 maio 2015. Citado 4 vezes nas páginas 51, 52, 78 e 96.

_____. 2016. Disponível em: <http://www.rcsb.org/pdb/static.do?p=software/software_links/molecular_graphics.html>. Acesso em: 06 maio 2016. Citado na página 52.

REYES, V. M. Representation of protein 3D structures in spherical (ρ, ϕ, θ) coordinates and two of its potential applications. **Interdisciplinary sciences, computational life sciences**, v. 3, p. 161–174, 2011. Citado na página 51.

RICHARDSON, J. S. The anatomy and taxonomy of protein structure. **Adv. Protein Chem.**, v. 34, p. 167–339, 1981. Citado na página 42.

ROBBETA.ORG. 2015. Disponível em: <http://www.robetta.org>. Acesso em: 25 jun. 2015. Citado na página 56.

ROSETTA. 2015. Disponível em: <https://www.rosettacommons.org>. Acesso em: 25 jun. 2015. Citado na página 57.

SAMPAIO, P. R. **Teoria, métodos e aplicações de otimização multiobjetivo**. Dissertação (Mestrado em Ciência da Computação) — Instituto de Matemática e Estatística - Universidade de São Paulo, São Paulo, 2011. Citado 2 vezes nas páginas 62 e 66.

SEELIGER, D.; GROOT, B. L. de. Atomic contacts in protein structures. A detailed analysis of atomic radii, packing, and overlaps. **Proteins**, Computational Biomolecular Dynamics Group, Max-Planck-Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany. bgroot@gwdg.de, v. 68, n. 3, p. 595–601, Aug 2007. ISSN 1097-0134. Citado na página 45.

SHIMADA, J.; SHAKHNOVICH, E. I. The ensemble folding kinetics of protein G from an all-atom Monte Carlo simulation. **Proceedings of the National Academy of Sciences**, v. 99, n. 17, p. 11175–11180, 2002. Disponível em: <http://www.pnas.org/content/99/17/11175.abstract>. Citado na página 84.

SIMONS, K. et al. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. **J. Mol. Biol.**, v. 268, n. 1, p. 209–25, Apr 1997. Citado na página 58.

SIPPL, M. J. Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. **J. Mol. Biol.**, v. 213, n. 4, p. 859–883, Jun 1990. Citado na página 60.

SKOLNICK, J. In quest of an empirical potential for protein structure prediction. **Curr. Opin. Struct. Biol.**, v. 16, n. 2, p. 166–171, Apr 2006. Citado na página 57.

SPOEL, D. van der et al. **Gromacs User Manual version 4.0**. [S.l.], 2009. Citado na página 95.

TALBI, E.-G. **Metaheuristics : from design to implementation**. [S.l.]: John Wiley & Sons, 2009. ISBN 9780470278581. Citado na página 66.

THREADER. 2015. Disponível em: <http://bioinf.cs.ucl.ac.uk/?id=747>. Acesso em: 20 jun. 2015. Citado na página 55.

TUFFERY et al. A new approach to the rapid determination of protein side chain conformations. **J. Biomol. Struct. Dyn.**, v. 8, n. 6, p. 1267–1289, 1991. Citado na página 68.

UCSF CHIMERA. 2015. Disponível em: <http://www.cgl.ucsf.edu/chimera>. Acesso em: 14 maio 2015. Citado na página 53.

UNIPROT. 2015. Disponível em: <http://www.ebi.ac.uk/uniprot/TrEMBLstats>. Acesso em: 25 jun. 2015. Citado na página 21.

VMD. 2015. Disponível em: <http://www.ks.uiuc.edu/Research/vmd>. Acesso em: 14 maio 2015. Citado na página 53.

WEINER, S. J. et al. A new force field for molecular mechanical simulation of nucleic acids and proteins. **Journal of the American Chemical Society**, v. 106, n. 3, p. 765–784, 1984. Disponível em: <http://dx.doi.org/10.1021/ja00315a051>. Citado na página 57.

WU, S.; SKOLNICK, J.; ZHANG, Y. Ab initio modeling of small proteins by iterative TASSER simulations. **BMC Biology**, v. 5, n. 1, p. 17, 2007. ISSN 1741-7007. Disponível em: <http://www.biomedcentral.com/1741-7007/5/17>. Citado na página 56.

WWPDB. 2015. Disponível em: <http://www.wwpdb.org>. Acesso em: 14 maio 2015. Citado na página 51.

XU, D.; ZHANG, Y. Improving the Physical Realism and Structural Accuracy of Protein Models by a Two-Step Atomic-Level Energy Minimization. **Biophysical Journal**, v. 101, p. 2525–2534, Nov. 2011. Citado na página 97.

YANG, Z. Protein structure prediction: when is it useful? **Current Opinion in Structural Biology**, v. 19, n. 2, p. 145–155, 2009. ISSN 0959-440X. Citado 3 vezes nas páginas 56, 57 e 58.

ZADEH, L. Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems. **IEEE Transactions on Automatic Control**, v. 8, n. 1, p. 59–60, Jan 1998. Citado na página 66.

ZELENY, M. Linear Multiobjective Programming. In: **Lecture Notes in Economics and Mathematical Systems**. 1. ed. Springer Berlin Heidelberg, 1974, (Methods Molecular Biology™, v. 95). p. 223. ISBN 978-3-642-80808-1. Disponível em: <http://www.springer.com/in/book/9783540066392>. Citado na página 65.

ZHANG, Y.; KIHARA, D.; SKOLNICK, J. Local energy landscape flattening: parallel hyperbolic Monte Carlo sampling of protein folding. **Proteins**, v. 48, n. 2, p. 192–201, Aug 2002. Citado na página 84.

ZHANG, Y.; SKOLNICK, J. Parallel-hat tempering: A Monte Carlo search scheme for the identification of low-energy structures. **J. Chem. Phys.**, v. 115, p. 5027–32, 2001. Citado na página 84.

ZWANZIG, R.; SZABO, A.; BAGCHI, B. Levinthal's Paradox. **Proc. Natl. Acad. Sci. USA**, v. 89, n. 1, p. 20–22, 1992. Citado na página 43.