Universidade de São Paulo
Escola de Engenharia de São Carlos
Departamento de Engenharia Elétrica
Programa de Pós-Graduação em Engenharia Elétrica

Rodrigo Antonio Faccioli

Implementação de um Framework de Computação Evolutiva Multi-Objetivo para Predição Ab Initio da Estrutura Terciária de Proteínas

São Carlos
2012

Rodrigo Antonio Faccioli

Implementação de um Framework de Computação Evolutiva Multi-Objetivo para Predição Ab Initio da Estrutura Terciária de Proteínas

Tese de doutorado apresentada ao Programa de Engenharia Elétrica da Escola de Engenharia de São Carlos como parte dos requisitos para a obtenção do título de Doutor em Ciências.

Área de concentração: Sistemas Dinâmicos

ORIENTADOR: Prof. Dr. Ivan Nunes da Silva

São Carlos
2012

Trata-se da versão corrigida da tese. A versão original se encontra disponível na EESC/USP que aloja o Programa de Pós-Graduação de Engenharia Elétrica.

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

F138i

Faccioli, Rodrigo Antonio.
Implementação de um *framework* de computação evolutiva multi-objetivo para predição *ab initio* da estrutura terciária de proteínas. / Rodrigo Antonio Faccioli ; orientador Ivan Nunes da Silva. -- São Carlos, 2012.

Tese (Doutorado - Programa de Pós-Graduação em Engenharia Elétrica e Área de Concentração em Sistemas Dinâmicos)-- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2012.

1. Algoritmos evolutivos multi-objetivo. 2. Predição da estrutura terciária de proteínas. 3. *Framework*. I. Título.

FOLHA DE JULGAMENTO

Candidato: Bacharel **RODRIGO ANTONIO FACCIOLI.**

Título da tese: "Implementação de um *framework* de computação evolutiva multi-objetivo para predição *ab initio* da estrutura terciária de proteínas".

Data da defesa: 24/08/2012

Comissão Julgadora:	**Resultado:**
Prof. Associado **Ivan Nunes da Silva (Orientador)** (Escola de Engenharia de São Carlos/EESC)	APROVADO
Prof. Dr. **Paulo Eduardo Ambrósio** (Universidade Estadual de Santa Cruz/UESC)	APROVADO
Prof. Associado **Alexandre Cláudio Botazzo Delbem** (Instituto de Ciências Matemáticas e de Computação/ICMC)	APROVADO
Prof. Dr. **Alexandre Suman de Araújo** (Universidade Estadual "Júlio de Mesquita Filho"/UNESP- São José do Rio Preto)	Aprovado
Prof. Titular **Antonio Caliri** (Faculdade de Ciências Farmacêuticas de Ribeirão Preto/USP)	APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia Elétrica e Presidente da Comissão de Pós-Graduação:
Prof. Titular **Denis Vinicius Coury**

Porque dele e por meio dele, e para ele, são todas as coisas. Glória, pois a ele eternamente. Amém

Romanos, 11, 36

Agradecimentos

Gostaria de agradecer a Deus por sempre me presentear com uma família maravilhosa e com pessoas amigas ao meu rendor, as quais contribuíram e muito para a elaboração deste trabalho. Procurarei agradecê-las com singelas palavras.

Agradeço a toda minha família pelo companheirismo, paciência e interesse em saber o andamento da minha pesquisa. À minha irmã, Renata, pela intensa atenção nos meus estudos. Ao meu sobrinho João Pedro, que mesmo com apenas três anos de idade, consegue com seu sorriso e, manhas as vezes, motivar-me a diminuir o cansaço. Ressalto meus sinceros agradecimentos a duas pessoas maravilhosas, as quais sinto muito orgulho em poder dizer: meus pais, Cláudio e Terezinha. Tenho que agradecê-los por toda dedicação nestes trinta anos.

O significado de família estende por avós, tios, tias, primos e primas. Assim, agradeço a todos pela atenção no que tange o progresso conquistado. Em especial gostaria de agradecer meus primos João Gabriel e Ana Maria por compartilhar seus momentos de lazer comigo.

Ao meu orientador prof. Dr. Ivan Nunes da Silva pelos ensinamentos não só em computação ou mesmo na escrita deste trabalho, mas também, nas lições de perseverança, amizade e disciplina.

Aos meus amigos do LAIPS onde tive a oportunidade de vivenciar o significado da palavra equipe. Em especial, gostaria deixar registrado meus agradecimentos ao meu grande amigo Marcelo Suetake, quem ajudou não só no template deste trabalho, mas também em valiosos ensinamentos e, em hipótese nenhuma posso deixar não evidenciar o seu companheirismo.

Aos meus amigos do laboratório de Física Biológica da FCFRP. Aqui gostaria de registrar os sinceros agradecimento ao Prof. Dr. Antonio Caliri por permitir não somente a realização deste trabalho em seu laboratório, mas também, da sua contribuição sobre o assunto *folding* de proteínas. Aos membros Ricardo, João, Flávio e Renata obrigado pela atenção, amizade e discussão dos resultados. Ao Guilherme agradeço pela contruibuição na implementação do algoritmo SN-Nerf. Um especial agradecimento ao Leandro, quem

acreditou na proposta deste trabalho e, assim, contribui com críticas, sugestões, scripts e explanações acerca de proteínas.

Em termos de acreditar nesta proposta, há a contribuição do Waldo Cancino Ticona. Mesmo morando na França, ele por meio do seu conhecimento do *framework* ParadisEO, vislumbramos o uso deste *framework*.

Ao meu grande amigo Túlio Calixto, agradeço pela sua amizade e atenção dada no andamento deste trabalho.

Aos membros da banca de qualificação desta Tese onde não só contribuíram tecnicamente, mas também, por enfatizar o significado da palavra foco. Neste sentido agradeço o prof. Dr. Alexandre Delbem pela sua contribuição na minha pesquisa desde o seu início e ao prof. Dr. Alexandre Suman de Araújo, quem por sua formação em física e computação, pode por meio do seu "pra que", fazer-me repensar em vários aspectos deste projeto. Inclusive, simplificando o nome do *framework* para 3PG.

À comunidade GROMACS que empregando a filosofia *open-source* tem-se um software que é competentemente empregado em modelagem molecular e se encontra em plena expansão.

Aos funcionários desta instituição, mas em especial a secretaria da pós-graduação da EESC, representada pela Marisa e Jussara. Além disso não posso deixar de agradecer a D. Vera, pois a fim de proporcionar momentos de discontração entre os alunos e professores, ela prepara um café apreciado por todos. Neste sentido, vale-se, também ser registrado, esse mesmo momento na FCFRP, onde a D. Evanira prepara o seu café recheado de assunto futebolístico.

Enfim, agradeço a todos que contribuíram na elaboração desta Tese, mas por um esquecimento não constam seus nomes. Mas mesmo assim, sua contribuição foi de igual significância, desta forma, além dos meus agradecimentos, fica registrado minhas sinceras desculpas.

Resumo

FACCIOLI, R. (2012). "Implementação de um Framework de Computação Evolutiva Multi-Objetivo para Predição Ab Initio da Estrutura Terciária de Proteínas". Tese de Doutorado – Escola de Engenharia de São Carlos, Universidade de São Paulo, 2012.

A demanda criada pelos estudos biológicos resultou para predição da estrutura terciária de proteínas ser uma alternativa, uma vez que menos de 1% das sequências conhecidas possuem sua estrutura terciária determinada experimentalmente. As predições *Ab initio* foca nas funções baseadas da física, a qual se trata apenas das informações providas pela sequência primária. Por consequência, um espaço de busca com muitos mínimos locais ótimos deve ser pesquisado. Este cenário complexo evidencia uma carência de algoritmos eficientes para este espaço, tornando-se assim o principal obstáculo para este tipo de predição. A otimização Multi-Objectiva, principalmente os Algoritmos Evolutivos, vem sendo aplicados na predição da estrutura terciária já que na mesma se envolve um compromisso entre os objetivos. Este trabalho apresenta o *framework* ProtPred-PEO-GROMACS, ou simplesmente 3PG, que não somente faz predições com a mesma acurácia encontrada na literatura, mas também, permite investigar a predição por meio da manipulação de combinações de objetivos, tanto no aspecto energético quanto no estrutural. Além disso, o 3PG facilita a implementação de novas opções, métodos de análises e também novos algoritmos evolutivos. A fim de salientar a capacidade do 3PG, foi então discorrida uma comparação entre os algoritmos NSGA-II e SPEA2 aplicados na predição *Ab initio* da estrutura terciária de proteínas em seis combinações de objetivos. Ademais, o uso da técnica de refinamento por Dinâmica Molecular é avaliado. Os resultados foram adequados quando comparado com outras técnicas de predições: Algoritmos Evolutivo Multi-Objetivo, *Replica Exchange Molecular Dynamics*, PEP-FOLD e Folding@Home.

Palavras-chave: Algoritmos Evolutivos Multi-Objetivo, Predição da Estrutura Terciária de Proteínas, *Framework*.

Abstract

FACCIOLI, R. (2012). "Implementation of Multi-Objective Evolutionary Framework for Ab Initio Protein Structure Prediction". Doctor Thesis – Engineering School of Sao Carlos, University of Sao Paulo, 2012.

The demand created by biological studies resulted the structure prediction as an alternative, since less than 1% of the known protein primary sequences have their 3D structure experimentally determined. *Ab initio* predictions focus on physics-based functions, which regard only information about the primary sequence. As a consequence, a search space with several local optima must be sampled, leading to insufficient sampling of this space, which is the main hindrance towards better predictions. Multi-Objective Optimization approaches, particularly the Evolutionary Algorithms, have been applied in protein structure prediction as it involves a compromise among conflicting objectives. In this paper we present the ProtPred-PEO-GROMACS framework, or 3PG, which can not only make protein structure predictions with the same accuracy standards as those found in the literature, but also allows the study of protein structures by handling several energetic and structural objective combinations. Moreover, the 3PG framework facilitates the fast implementation of new objective options, method analysis and even new evolutionary algorithms. In this study, we perform a comparison between the NSGA-II and SPEA2 algorithms applied on six different combinations of objectives to the protein structure. Besides, the use of Molecular Dynamics simulations as a refinement technique is assessed. The results were suitable when comparated with other prediction methodologies, such as: Multi-Objective Evolutionary Algorithms, Replica Exchange Molecular Dynamics, PEP-FOLD and Folding@Home.

Keywords: Multi-Objective Evolutionary Algorithms, *Ab initio* Protein Structure Prediction, Framework.

Lista de Ilustrações

2.1 Estrutura básica de um aminoácido. 10
2.2 Classificação dos vinte aminoácidos padrões encontrados em proteínas. 11
2.3 Processo de formação de uma ligação peptídica. 12
2.4 Ângulos ψ e ϕ. 13
2.5 Mapa de Ramachandran. 14
2.6 Estrutura Hélice α. 15
2.7 Estrutura Folhas β. 15
2.8 Representação da estrutura terciária de uma proteína (PDB 1CCN). 16
2.9 Representação do ângulo torsional τ. 17
2.10 Representação do perfil energético de *folding* por meio do funil (DILL et al., 2008). O N representa a estrutura Nativa. 19

3.1 Exemplo do multi-objetivo (TICONA, 2003). 32
3.2 Esquema do Modelo NSGA-II (DEB, 2001). 41
3.3 Cálculo da distância de multidão no NSGA-II (DEB, 2001). 41

4.1 Representação do Diagrama do *framework* proposto. 51

5.1 Representação do arquivo Fasta da proteína 1VII. 54
5.2 Representação da Seção **General Information** da proteína 1VII. 54
5.3 Representação da topologia do 3PG referente as seções **Atom** e **Residue and its sequence atoms proteína** 1VII. 55
5.4 Representação da topologia do 3PG referente as seções ϕ, ψ e *chi* da proteína 1VII. 56
5.5 Representação da matriz Z para a proteína 1VII 57
5.6 Representação do posicionamento de átomo pelo SN-Nerf. Figura baseada em (PARSONS et al., 2005). 59
5.7 Representação da conversão da estrutura nativa da proteína 1PLW aplicando o algoritmo SN-Nerf e a topologia do 3PG. 59

5.8 Representação da conversão da estrutura nativa da proteína 1A11 aplicando o algoritmo SN-Nerf e a topologia do 3PG. 60
5.9 Representação da conversão da estrutura nativa da proteína 1UAO aplicando o algoritmo SN-Nerf e a topologia do 3PG. 60
5.10 Representação da conversão da estrutura nativa da proteína 1VII aplicando o algoritmo SN-Nerf e a topologia do 3PG. 61
5.11 Diagrama UML da classe ProteinMOEO a qual representa a solução (Proteína). Esta é derivada da classe MOEO provida pelo componente ParadisEO-MOEO do ParadisEO. Assim, é possível visualizar informações a cerca da organização interna do ParadisEO no que tange a sua representação da solução. 63
5.12 Representação do diagrama UML da classe ProteinInit que herda a classe eoInit do ParadisEO. Esta nova classe tem a finalidade de incorporar ao ParadisEO a população inicial. Os detalhes da criação da população inicial encontram-se na Subseção 5.1. 64
5.13 Diagrama UML da classe ProteinMOEO_TorsionAngles_Crossover a qual foi herdada da classe eoQuadOp do ParadisEO. Esta nova classe representa a integração do operador genético crossover do protpred-GROMACS no ParadisEO. 64
5.14 Diagrama UML da classe ProteinMOEO_TorsionAngles_Mutation a qual foi herdada da classe eoMonOp do ParadisEO. A classe herdada representa a integração do operador genético de mutação do protpred-GROMACS no ParadisEO. 65
5.15 Representação do diagrama UML da computação dos *fitness* realizado pelo protpred-PEO-GROMACS. A classe ProteinMOEOPopEval enfatiza a integração do algoritmo protpred com o GROMACS. Assim, esta classe herda a classe eoPopLoopEval que por sua vez herda da classe eoPopEvalFunc. Estas duas últimas classes pertencem ao *framework* ParadisEO. 65
5.16 Representação das fronteiras de Pareto finais referentes à predição do peptídeo 1PLW pelos algoritmos NSGA-II e SPEA2 em várias combinações de objetivos. 68
5.17 Representação das images das estruturas do peptídeo 1PLW, respectivamente: nativa, melhor RMSD não-refinado (Etapa I) e melhor RMSD refinado (Etapa II). 70
5.18 Representação das fronteiras de Pareto finais referentes à predição do peptídeo 1UAO pelos algoritmos NSGA-II e SPEA2 em várias combinações de objetivos. 71
5.19 Ilustração da comparação da estrutura nativa e cada uma das etapas da predição pelo 3PG do peptídeo 1UAO. 73
5.20 Representação das fronteiras de Pareto finais referentes à predição da proteína 1VII pelos algoritmos NSGA-II e SPEA2 em várias combinações de objetivos. 74
5.21 Ilustração da comparação da estrutura nativa e cada uma das etapas da predição pelo 3PG da proteína 1VII. 76

Lista de Tabelas

2.1 Relação dos vinte aminoácidos padrões e respectivos mnemônicos. 10

3.1 Alguns exemplos de modelos de MOEA. 40

5.1 RMSDs finais (Å) calculados utilizando a energia Potencial como *fitness*. O número de gerações foi 1000. 62

5.2 RMSDs finais (Å) calculados utilizando a área hidrofóbica acessível ao solvente como *fitness*. O número de gerações foi 1000. 62

5.3 Melhores RMSDs Å obtidos pelo ProtPred-GROMACS e o seu valor correspondente de RMSD nas predições encontradas na literatura. 62

5.4 Valores dos RMSDs obtidos na predição do peptídeo 1PLW na etapa de exploração do espaço de busca. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD não-refinado e a segunda linha é o melhor RMSD final não-refinado. Todos os valores estão em Å. 68

5.5 Valores dos RMSDs obtidos na predição do peptídeo 1PLW na etapa de refinamento estrutural. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD refinado e a segunda linha é o melhor RMSD final refinado. Todos os valores estão em Å. 69

5.6 Melhores RMSDs em termos C-α, *backbone* e *all atom* para o peptídeo 1PLW. Todos os valores estão em Å. 69

5.7 Valores dos RMSDs obtidos na predição do peptídeo 1UAO na etapa de exploração do espaço de busca. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD não-refinado e a segunda linha é o melhor RMSD final não-refinado. Todos os valores estão em Å. 71

5.8 Valores dos RMSDs obtidos na predição do peptídeo 1UAO na etapa de refinamento estrutural. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD refinado e a segunda linha é o melhor RMSD final refinado. Todos os valores estão em Å. 72

5.9 Melhores RMSDs em termos C-α, *backbone* e *all atom* para o peptídeo 1UAO. Todos os valores estão em Å. 72

5.10 Valores dos RMSDs obtidos na predição da proteína 1VII na etapa de exploração do espaço de busca. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD não-refinado e a segunda linha é o melhor RMSD final não-refinado. Todos os valores estão em Å. 74
5.11 Valores dos RMSDs obtidos na predição da proteína 1VII na etapa de refinamento estrutural. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD refinado e a segunda linha é o melhor RMSD final refinado. Todos os valores estão em Å. 75
5.12 Melhores RMSDs em termos C-α, *backbone* e *all atom* para a proteína 1VII. Todos os valores estão em Å. 75

Lista de Siglas

3PG ProtPred-PEO-GROMACS

AE Algoritmo Evolutivo

AG Algoritmo Genético

CASP *Critical Assessment of methods of Protein Structure Prediction*

CE Computação Evolutiva

MOEA *Multi-Objective Evolutionary Algorithm*

NSGA-II *Elitist Non-Dominated Sorting Genetic*

POMO Problemas de Otimização Multi-Objetivo

PSP *Protein Structure Prediction*

SN-Nerf *Self-Normalizing Natural Extension Reference Frame*

SPEA2 *Strenght Pareto Evolutionary Algorithm 2*

Sumário

1 Introdução e Motivação **1**
1.1 Motivação e Relevância do Trabalho 4
1.2 Objetivos e Contribuições da Presente Pesquisa 6
1.3 Organização da Tese 8

2 Fundamentos sobre Proteínas **9**
2.1 Aminoácidos 9
2.2 Ligações Peptídicas e os Polipeptídeos 12
2.3 Estrutura Primária de Proteína 13
2.4 Estrutura Secundária de Proteína 14
2.5 Estrutura Terciária de Proteína 15
2.6 Representação Computacional da Proteína 17
2.7 O Problema do *Folding* de Proteínas 18
2.8 Interações da Proteína 24
2.9 Considerações Parciais 28

3 Computação Evolutiva **31**
3.1 Otimização Multi-Objetivo 31
3.2 Teoria da Evolução 33
3.3 Algoritmos Genéticos 34
3.4 Algoritmo Evolutivo Multi-objetivo 39
3.5 Considerações Parciais 43

4 Metodologia Proposta **45**
4.1 Apresentando o ProtPred 45
4.2 Modificando o ProtPred 48
4.3 Considerações Parciais 51

5 Resultados e Aspectos de Desenvolvimento do 3PG **53**
5.1 Integração do ProtPred com GROMACS 53

5.2 Avaliação da Conversão de Coordenadas Internas para Cartesianas 58
5.3 Aplicação do ProtPred-GROMACS no PSP 61
5.4 Integração do ProtPred-GROMACS com ParadisEO 62
5.5 Aplicação do ProtPred-PEO-GROMACS no PSP 65

6 Conclusões **77**
6.1 Trabalhos Futuros 78

Referências **79**

Apêndices **93**

A Cálculo das Propriedades da Proteína **95**
A.1 Energia Potencial 95
A.2 Raio de Giro 97
A.3 Área da Superfície de Acessibilidade do Solvente 97
A.4 Ligações de Hidrogênios 97

Capítulo 1

Introdução e Motivação

Proteínas são macromoléculas biológicas envolvidas em uma variedade de funções, como sinalização e reconhecimento celular, nas quais estrutura e função estão intimamente relacionadas. Assim, o conhecimento do arranjo tridimensional dos resíduos que as compõem fornece informações valiosas para o melhor etendimento de suas propriedades bioquímicas e funções biológicas (CREIGHTON, 1992).

Em modelagem molecular, a determinação do arranjo dos átomos da molécula que correspondem ao seu estado de mínimo de energia fundamenta-se como uma tarefa imprescindível, pois é nesta conformação em que a molécula se encontra na maior parte do tempo e, assim, pode realizar a sua função. Ela é também conhecida como conformação nativa. No entanto, ainda não há um método computacional eficiente para realizar tal tarefa. Em outras palavras, ainda não foi possível provar a eficiência de um algoritmo em localizar o mínimo de energia potencial a partir de uma inicialização arbitrária de átomos (LEACH, 2001).

Em virtude da complexidade e heterogeneidade envolvida nos problemas de modelagem molecular, torna-se evidente um desenvolvimento computacional substancial para atendê-los. Neste sentido, a literatura reporta alguns *frameworks*: GROMACS (HESS et al., 2008), Faunus (LUND; TRULSSON; PERSSON, 2008), Tinker (PONDER, 2001), etc. O GROMACS vem se destacando por duas de suas características principais: a acurácia e a rapidez de computar as propriedades físicas de proteínas. Além disso, o GROMACS é um projeto *open-source* e, assim, consegue rapidamente se expandir com agregações de novas implementações, uma vez que há contribuições de vários pesquisadores. Em SPOEL; HESS (2011) é destacado as novas implementações e, acima de tudo, os rumos desse projeto o qual atende as exigências da área de modelagem molecular no que tange computação das propriedades físicas de proteínas.

Realmente, um dos principais tópicos do problema de *folding* de proteínas, e de outros polímeros, é a questão de como predizer a conformação nativa somente possuindo a sequência de seus aminoácidos (DILL et al., 2008) . Além disso, o mínimo global pode não representar a conformação nativa da molécula (conformação biologicamente ativa),

mas é sabido que pelo menos um de seus mínimos corresponderá a esse estado (LEACH, 2001). Logo, esse problema necessita de tratamento computacional eficiente em virtude de sua complexidade (NAIR; GOODMAN, 1998).

O *folding* de proteínas é um problema computacional complexo. Segundo a conjectura de Levinthal, se as proteínas procurassem aleatoriamente todas as conformações possíveis até encontrar a de menor energia, o tempo necessário para esta tarefa seria então maior que o tempo correspondente à idade do universo. Logo, as proteínas devem empregar caminhos de *fold*[1] peculiares que evitam extensivas buscas no espaço conformacional (LEVINTHAL, 1968).

Em KARPLUS; SHAKHNOVICH (1992) são descritas algumas propriedades essenciais no *folding*, entre elas, a ligação do *backbone* ao longo da sequência, interações entre os aminoácidos (incluindo as forças electrostática e van der Waals), restrições de volume, ligação de hidrogênio e ligações químicas entre as cisteínas, além das interações da cadeia com a água.

Essas situações em que há a necessidade de se encontrar o extremo (mínimo ou máximo) de uma função são conhecidas como problemas de otimização. Estes podem ser divididos em dois principais métodos: (i) Exato e (ii) Aproximação (ou Heurístico). O primeiro consegue garantir a obtenção das melhores soluções, ou seja, é possível encontrar o mínimo/máximo da função, mas o tempo computacional pode ser extremamente alto. Já no segundo método, as soluções com alta qualidade podem ser alcançadas, com a vantagem de serem obtidas em um tempo computacional reduzido, embora não haja garantia de que as soluções sejam de mínimo global. Este último método pode ser classificado em duas famílias: heurística específica e meta-heurística. Heurísticas específicas são empregadas em problemas bem particulares. Já as meta-heurísticas são aplicadas em propostas gerais e, assim, podem resolver uma variedade de problemas diferentes por meio da mesma estratégia (TALBI, 2009).

Uma outra característica dos problemas de otimização é a sua função objetivo, a qual pode ser mono-objetivo ou multi-objetivo. A mono-objetivo contém apenas um único objetivo. Porém, existe uma gama de problemas do mundo real os quais envolvem a análise simultânea de mais que um único objetivo. Tais problemas podem ser modelados sob a óptica do mono-objetivo. Para tanto, é atribuído um valor, conhecido como peso, para cada um dos seus objetivos baseando-se na sua importância no problema em questão. Estes valores são normalmente atribuídos de acordo com o conhecimento que se possui do problema. Por outro lado, há problemas de otimização cujo conhecimento sobre o mesmo não é suficiente para atribuir tais valores. Existem também casos em que não há como atribuir os pesos, pois a análise de cada um dos objetivos exige ser realizada separadamente. Nestes dois últimos casos, em que a atribuição dos valores de pesos não é eficaz, exige-se o emprego da técnica multi-objetivo (DEB, 2001).

[1] Padrões Estruturais.

Dentre as técnicas computacionais as quais vêm sendo empregadas nestes tipos de problemas, destacam-se os Algoritmos Evolutivos (AE), sendo mais representado pelo Algoritmo Genético (AG). Estes são uma meta-heurística inspirada na teoria da evolução (GOLDBERG, 1989) sendo os mesmos aplicados em virtude de sua capacidade em explorar o espaço de busca, aproveitando as melhores soluções (MICHALEWICZ; SCHOENAUER, 1996). Além disso, os AEs são fáceis de empregar nos problemas de otimização multi-objetivo e mono-objetivo (JAIMES; COELLO, 2008). A primeira implementação dos Algoritmos Evolutivos Multi-Objetivos (MOEA, do inglês *Multi-Objetive Evolutionary Algorithms*) foi proposta por Schaffer em 1985 (SCHAFFER, 1985a). Nesta implementação foi realizada uma modificação do AG convencional em que seus objetivos foram analisados de forma independente.

Em virtude dos AEs serem empregados em diversos tipos de problemas, a literatura vem reportando o desenvolvimento de *frameworks*, os quais implementam os conceitos genéricos dos AEs (técnicas de seleção, operadores genéticos, Dominância de Pareto por exemplo), para assim obter uma padronização dos AEs empregados: TEA (EMMERICH; HOSENBERG, 2001), BEAGLE (GAGNE; PARIZEAU; DUBREUIL, 2003), PISA (BLEULER et al., 2003), ParadisEO (LIEFOOGHE et al., 2007), MALLBA (ALBA et al., 2007), Shark (IGEL; GLASMACHERS; HEIDRICH-MEISNER, 2008), JMetal (DURILLO; NEBRO, 2011), etc. Dentre estes, destaca-se o ParadisEO (LIEFOOGHE et al., 2007), um *framework* para a computação evolutiva que permite trabalhar com os AEs mono-objetivo e multi-objetivo.

Os AEs vêm sendo também aplicados no campo da biologia. O primeiro trabalho relacionado à aplicação dos AEs para analisar a conformação de proteínas foi aquele de MCGARRAH; JUDSON (1993), no qual se seguiram outros trabalhos como WENG et al. (2005) e LIU et al. (2009). Neste sentido há trabalhos que enfatizam sua modelagem sob a óptica multi-objetivo (HANDL; KELL; KNOWLES, 2007); (JAIMES; COELLO, 2008), como é o caso do Problema de Predição de Estrutura Terciária de Proteínas (PSP, do inglês *Protein Structure Prediction*).

Mais especificamente, no PSP, a aplicação dos AEs vem aumentando recentemente (DAY; LAMONT; PACHTER, 2003); (LIMA et al., 2007); (FACCIOLI et al., 2011); (FACCIOLI et al., 2012). Segundo CUTELLO; NARZISI; NICOSIA (2006a) existe uma competição entre as propriedades físicas acima descritas para o PSP. Neste sentido, o trabalho de HANDL; LOVELL; KNOWLES (2008) investigou com mais ênfase a abordagem multi-objetivo destas interações. Assim, concluíram-se que o PSP é um caso apropriado para os MOEAs, sendo então exibidos na literatura trabalhos aplicando MOEAs para o PSP, como em CUTELLO; NARZISI; NICOSIA (2006a), LIMA (2006), HANDL; KELL; KNOWLES (2007), CALVO et al. (2009), BRASIL; DELBEM; BONETTI (2011), MÁRQUEZ-CHAMORRO et al. (2012).

1.1 Motivação e Relevância do Trabalho

A demanda criada pelos estudos biológicos atribui à predição de estruturas terciária de proteínas uma alternativa, uma vez que é estimado que menos de 1% das sequências primárias conhecidas tem-se a sua estrutura terciária (YANG; ZHANG, 2009). Uma das principais contribuições do uso dos algoritmos de predição de estruturas terciárias de proteínas, tendo como *input* a sequência de aminoácidos, é a realização do *drug discovery* mais rápido e eficiente, uma vez que tornará possível integrar os caros e demorados experimentos biológicos estruturais com a simulação computacional que é mais rápida e barata (DILL et al., 2007).

COHEN (2004) evidencia a importância do cientista da computação para auxiliar os biólogos a interpretar o volumoso conjunto de dados oriundos da pesquisa genômica e proteômica, além de desenvolver *in silício* aquilo que serão integrados (utilizados) em experimentos *in vivo* e *in vitro*. Sendo assim, por intermédio de *data mining*, é também possível identificar determinados padrões em uma grande quantidade de dados a partir de algoritmos de aprendizagem. Como exemplo, as redes neurais artificiais foram usadas como ferramentas de *data mining* para predizer a ocorrência de câncer de mama (PENDHARKAR et al., 1999; CHOU et al., 2004). Dentre as aplicações de MOEAs para predizer a estrutura terciária de proteínas destacam-se os trabalhos de CUTELLO; NARZISI; NICOSIA (2006a), LIMA (2006), HANDL; KELL; KNOWLES (2007), CUTELLO; NARZISI; NICOSIA (2008), CALVO et al. (2009) e BRASIL; DELBEM; BONETTI (2011).

ECHENIQUE (2007) apresenta uma classificação dos métodos de predição de estrutura de proteína (modelagem comparativa, *Threading* e *Ab initio*) nos requisitos "dados experimentais" e "princípios físicos". Para Echenique, quanto mais próximo o método depende dos dados experimentais, menos necessita de poder computacional. Entretanto, por outro lado, quanto mais próximo o método está do princípio físico, em virtude de não possuir uma larga gama de informação, este então necessita de um poder computacional elevado, para assim tentar suprir essa carência. Nesta classificação, a modelagem comparativa, possuindo maiores informações da estrutura, não exige um poder computacional substancial para a tarefa de predizer estruturas. Por outro lado, o método *Ab initio* exige o maior poder computacional. Portanto, é conclusivo que quanto mais informações se tem a respeito da estrutura, mais fácil ficará a predição da mesma.

De fato, o campo de biologia molecular, assim como da modelagem molecular, é feito sob medida para os métodos baseados em sistemas inteligentes, visto que tais métodos têm uma ótima performance onde há muitos dados quantitativos, mas pouca informação qualitativa disponível (EZZIANE, 2006). Desde a introdução de técnicas de sistemas inteligentes nesta área, muitos algoritmos têm sido propostos e aplicados para o estudo de diferentes grupos de dados. Mais especificamente, o PSP sendo modelado como um problema de otimização possui um espaço de busca com rugosidade (descontínuo, irregular, multi-modal, não-linear e ruidoso), onde se justifica o uso de algoritmos meta-heurísticos,

uma vez que os algoritmos baseados em derivadas, tais como *Conjugate Gradient*, não se obtem êxito (TALBI, 2009).

Por sua vez, os AEs, pertencendo à classe de algoritmos meta-heurísticos, têm já sido também utilizados de forma extensiva como uma metodologia de otimização na solução de diversos problemas envolvendo à predição de estruturas terciárias de proteínas. Neste sentido, os algoritmos evolutivos são métodos de otimização adaptativos que se utilizam operadores, os quais são inspirados em métodos seletivos naturais, que são especialistas na procura de soluções ótimas. Prova-se que os métodos baseados nos algoritmos evolutivos são teoricamente e empiricamente robustos em espaços complexos (DEB, 2001; DEJONG, 2006). Assim, 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. Consequentemente, as abordagens inspiradas nos algoritmos evolutivos podem ser aplicadas em diversos problemas envolvidos com a predição de estruturas terciárias de proteínas, nas quais requerem a procura no espaço de conformações, levando-se em consideração os critérios ótimos associados com os potenciais de energia e estruturais.

Para serem computados os potenciais de energia e estruturais deve-se então obter valores das propriedades físicas das proteínas. Para isto, é necessário entender a interação dos resíduos, átomos e grupo de átomos da proteína e como eles se interagem com o meio. Tendo o objetivo de parametrizar as equações matemáticas dessa interação, os campos de forças foram desenvolvidos (LEACH, 2001). Existem, na literatura, inúmeros campos de forças, os quais podem ser específicos ou especializados em alguma molécula ou propriedades. São exemplos de campos de forças: OPLS, GROMOS96, Martini e o CHARMM. A tarefa mais árdua da computação (na prática) de modelagem molecular é a obtenção dos parâmetros de campo de força, tipos de átomo, e outras informações necessárias para a obtenção das propriedades físicas das proteínas (SPOEL; HESS, 2011).

De fato, o campo da modelagem molecular é cada vez mais útil nas pesquisas básicas tal como a biotecnologia. Entretanto, a ausência de um *framework user-friendly*, o qual providencia um acesso a uma gama de informações moleculares, vem descorajando a sua adoção de métodos computacionais por não especialistas em computação, os quais possuem muita experiência na área experimental (SAREL et al., 2011). Todos os *frameworks* proporcionam algumas informações acerca de proteínas, mas não todas. Portanto, é necessário trabalhar com mais de um *framework*. tornando-se aqui uma situação desfavorável já que os mesmos podem trabalhar com diferentes campos de forças e/ou unidades.

Mais especificamente, o PSP sendo um problema complexo e sem uma solução plausível, os pesquisadores vêm então aplicando diferentes técnicas e métodos. Para este cenário, torna-se evidente a adoção de *frameworks* especializados para o PSP. Neste sentido, o trabalho de HONIG (1999) buscou o desenvolvimento do PrISM, uma plataforma computacional com uma completa integração sequência/análise estrutural/*fold-recognition*/homologia. Já em KLEPEIS; FLOUDAS (2003); SUBRAMANI; WEI; FLOU-

DAS (2012) enfatiza-se um *framework* chamado ASTRO-FOLD para abordagem de predição *ab initio*. O projeto ProteinShop (CRIVELLI et al., 2004) fundamenta-se em uma ferramenta interativa para manipulação das estruturas proteicas. Ela é designada para criar rapidamente uma diversidade de estrutuas iniciais tendo uma sequência de aminoácido. No *Critical Assessment of methods of Protein Structure Prediction* (CASP), comunidade mundial em que cada biênio realiza uma competição sobre os algoritmos de predição, vem se destacando dois algortimos: o Rosetta (SIMONS et al., 1999) e o I-TASSER (ZHANG, 2008a). Em tais trabalhos somente é possível modificar os parâmetros da técnica proposta. Ou seja, estes utilizam um único algoritmo, o qual simplesmente recebe o alvo, executa e devolve as estrutuas terciárias. Assim, não se pode alavancar o desenvolvimento de novos algoritmos.

1.2 Objetivos e Contribuições da Presente Pesquisa

A finalidade deste trabalho é o desenvolvimento do ProtPred-PEO-GROMACS (3PG), um *framework* de computação evolutiva para o PSP. Neste *framework*, a abordagem evolutiva enfatiza-se pelo uso do ParadisEO (PEO) e a obtenção das propriedades da proteína é realizada pelo GROMACS. Já o ProtPred tem a responsabilidade de integrar ambos *frameworks* de maneira *user-friendly* tanto para a execução dos algoritimos para o PSP quanto para a análise de sua performance.

Os trabalhos de LIMA (2006); LIMA et al. (2007) desencadearam o desenvolvimento do *software* chamado ProtPred, recentemente publicado em BRASIL; DELBEM; BONETTI (2011), é aplicado no PSP modelando-o como um problema de otimização mono e multi-objetivo. Para a computação das propriedades da proteína é utilizado implementações baseadas no Tinker (PONDER, 2001). Vale ressaltar que o 3PG calcula essas propriedades por meio da integração com o GROMACS. Além disso, no *framework* proposto é utilizado a implementação do algoritmo SN-Nerf (*Self-Normalizing Natural Extension Reference Frame*) (PARSONS et al., 2005) para a conversão da representação da proteína em espaço diedral para o cartesiano. Outra diferença importante é a construção da topologia atomística da proteína visando uma melhora de desempenho computacional.

A motivação do desenvolvimento desta pesquisa fundamentou-se da carência de um ambiente de simulação e testes de metodologias para o PSP integrado. Em outras palavras, não há em um único ambiente computacional a possibilidade de investigar modelos físicos com algoritmos populacionais, por exemplo, o AE. Ou seja, os AEs propostos são oriundos de outras aplicações e, assim, não há uma padronização de testes. Logo, o que se busca neste projeto também é a integração dos AEs com uma plataforma de simulação molecular.

Em CUTELLO; NARZISI; NICOSIA (2006a) se desenvolveu o MOEA chamado I-PAES e o aplicou no PSP. O I-PAES é uma modificação do MOEA *Pareto Archived Evolution Strategy* (PAES)(KNOWLES; CORNE, 1999) a qual se baseou na introdução

dos operadores genéticos propostos em CUTELLO et al. (2004). Desta forma, o *framework* proposto também permitirá uma infraestrutura para o desenvolvimento de operadores específicos para o PSP.

Tendo o objetivo de alavancar uma gama de informações acerca das propriedades da proteína e, assim, empregá-las nos algoritmos evolutivos é, então, utilizado o *framework* GROMACS. As propriedades são:

1. Energia Potencial.
2. Eletrostática.
3. Leonard Jones.
4. Raio de Giro.
5. Ligações de Hidrogênio.
6. Área de acessibilidade com o solvente, sendo as opções:
   - ❏ Hidrofóbica;
   - ❏ Hidrofílica;
   - ❏ Área total.

O ParadisEO, sendo um *framework* para a Computação Evolutiva (CE), encontra-se em pleno desenvolvimento com novas características, as quais visam intensificar e facilitar a investigação da CE em problemas de otimização. Já a aplicação da CE no PSP necessita de tratamentos, os quais é o objetivo do 3PG. Tais tratamentos estão sistematizados nos seguintes itens:

- ❏ Uso do GROMACS para obtenção das propriedades da proteína.
- ❏ Emprego do algoritmo SN-Nerf (PARSONS et al., 2005) na conversão de coordenadas Internas (espaço diedral) em coordenadas Cartesianas (espaço cartesiano).
- ❏ Construção da topologia atomística da proteína baseando-se na sua sequência de aminoácidos contendo as informações: átomos e seu aminoácido, bem como a sua carga e os quatro átomos que formam os ângulos diedrais do *backbone* e cada tipo de cadeia lateral dos aminoácidos.

A pesquisa proposta quando comparada com os *frameworks* especializados no PSP, principalmente os trabalhos já citados em HONIG (1999) e KLEPEIS; FLOUDAS (2003); SUBRAMANI; WEI; FLOUDAS (2012), qualifica-se como uma tendência promissora, pois ela é flexível no que tange empregar diferentes algoritmos na mesma plataforma. Consequentemente, os pesquisadores podem trabalhar em um único *framework*.

Além disso, neste trabalho, enfatiza-se o uso de técnicas populacionais para a modelagem molecular, as quais não são utilizadas pelos *framework* citados. Tais algoritmos, embora seu uso vem crescendo, não possuem um únco ambiente de simulação integrado com os programas de modelagem molecular. Por fim, mais especificamente para o PSP, os algoritmos evolutivos multi-objetivo vem sendo difundido seu uso, pois no PSP, envolve um compromisso entre os diferentes objetivos. Desta forma, uma solução ótima em um dos objetivos pode não ser ótima no outro (CUTELLO; NARZISI; NICOSIA, 2006a); (HANDL; LOVELL; KNOWLES, 2008).

Diante deste cenário, o 3PG está pautado na tentativa de suprir a carência da amostragem do espaço de busca das predições *ab initio* (ZHANG, 2008b), pois o mesmo integrado com *frameworks* de modelagem molecular e algoritmos populacionais torna-se possível o desenvolvimento de novos preditores de forma ágil e padronizada.

1.3 Organização da Tese

O Capítulo 2 tratará do estudo teórico do problema de predição de estruturas terciárias de proteínas. Contemplar-se-á desde a definição de proteínas, aminoácidos e ligações peptídicas até a determinação da estrutura de proteínas a partir de sua sequência de aminoácido.

O Capítulo 3 abordará os aspectos teóricos sobre o problema de otimização multi-objetivo e a computação evolutiva. Enfatizar-se-á a descrição dos dois algoritmos evolutivos multi-objetivo empregados.

O Capítulo 4 é referente à metodologia proposta, além de ser possível verificar o cenário de aplicação do *framework* apresentado.

O Capítulo 5 evidenciará os resultados da aplicação do *framework* proposto para o problema de predição da estrutura terciária da proteína.

O Capítulo 6 descreverá as conclusões deste trabalho de doutorado.

Capítulo 2

Fundamentos sobre Proteínas

Proteínas são hétero-polímeros cujas unidades são constituídas a partir de um alfabeto de 20 aminoácidos. Uma importante propriedade das proteínas é a classificação de suas estruturas hierarquicamente: estrutura primária, estrutura secundária e estrutura terciária (tridimensional).

Uma das relevâncias em investigar as proteínas é que as mesmas são macromoléculas que desempenham as mais diversas funções no organismo. Para que consigam exercer sua função biológica, elas devem estar em uma conformação tridimensional bem definida, chamada de estrutura nativa. Proteínas podem ligar-se seletivamente a outras macromoléculas, tais como DNA, carboidratos, ou outras proteínas. Esta habilidade é decorrente do fato das proteínas apresentarem superfícies estruturalmente e quimicamente diversas, possibilitando a interação com outras moléculas com alta especificidade (PETSKO; RINGE, 2004).

É conhecido como *folding* ou enovelamento, o processo em que um hetero-polímero (proteína, por exemplo) encontra sua estrutura nativa. Embora seja um processo extremamente importante, o mesmo ainda não é completamente entendido. Existem diversas metodologias as quais buscam predizer a estrutura nativa possuindo somente a estrutura primária. Dentre elas há a abordagem de primeiros princípios, onde as interações de nível atomístico são empregadas.

2.1 Aminoácidos

Aminoácidos são compostos orgânicos que possuem uma estrutura básica comum que consiste de um carbono central, denominado carbono α, o qual possui quatro ligantes diferentes: um hidrogênio (H), um grupo carboxila (COOH), um grupo amina (NH_2) e um radical R, também chamado cadeia lateral do aminoácido (que pode consistir de um único átomo de hidrogênio até complexos anéis aromáticos) (COPELAND, 1993). A Figura 2.1 representa a estrutura básica de um aminoácido.

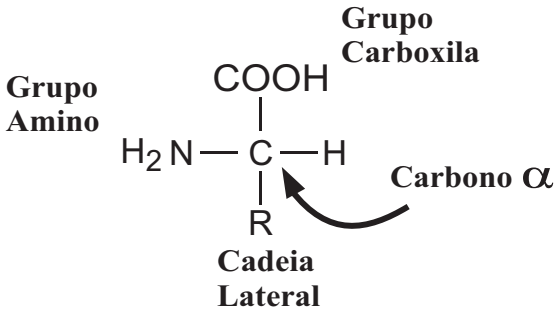


Figura 2.1: Estrutura básica de um aminoácido.

As proteínas são formadas a partir de um conjunto de vinte aminoácidos que se diferenciam por suas cadeias laterais. Os aminoácidos presentes em proteínas são denominados "resíduos", pois no processo de formação da proteína ocorre a perda de átomos (uma molécula de água - H_2O) que compunham a estrutura completa do aminoácido.

Tanto um código de três, ou de apenas uma letra, são utilizados frequentemente para representar os aminoácidos ou resíduos. Os vinte aminoácidos estão apresentados na Tabela 2.1, com seus respectivos códigos, bem como seu peso molecular. A Figura 2.2 apresenta a estrutura química dos vinte aminoácidos presentes nas proteínas.

Tabela 2.1: Relação dos vinte aminoácidos padrões e respectivos mnemônicos.

Aminoácido	Cód. 3 letras	Cód. 1 letra	Origem do Cod. 1 Letra	Peso Molecular
Alanina	Ala	A	*A*lanine	71
Cisteína	Cys	C	*C*ysteine	103
Ácido Aspártico	Asp	D	aspar*D*ic acid	114
Fenilalanina	Phe	F	*F*enylalanine	147
Ácido Glutâmico	Glu	E	glu*E*tamic acid	128
Glicina	Gly	G	*G*lycine	57
Histidina	His	H	*H*istidine	137
Isoleucina	Ile	I	*I*soleucine	113
Lisina	Lys	K	letra antes do *L*	129
Leucina	Leu	L	*L*eucine	113
Metionina	Met	M	*M*ethionine	131
Asparagina	Asn	N	asparagi*N*e	114
Prolina	Pro	P	*P*roline	97
Glutamina	Gln	Q	*Q*-tamine	128
Arginina	Arg	R	a*R*ginine	157
Serina	Ser	S	*S*erine	87
Treonina	Thr	T	*T*heorine	101
Valina	Val	V	*V*aline	99
Triptofano	Trp	W	t*W*o rings	186
Tirosina	Tyr	Y	t*Y*rosine	163

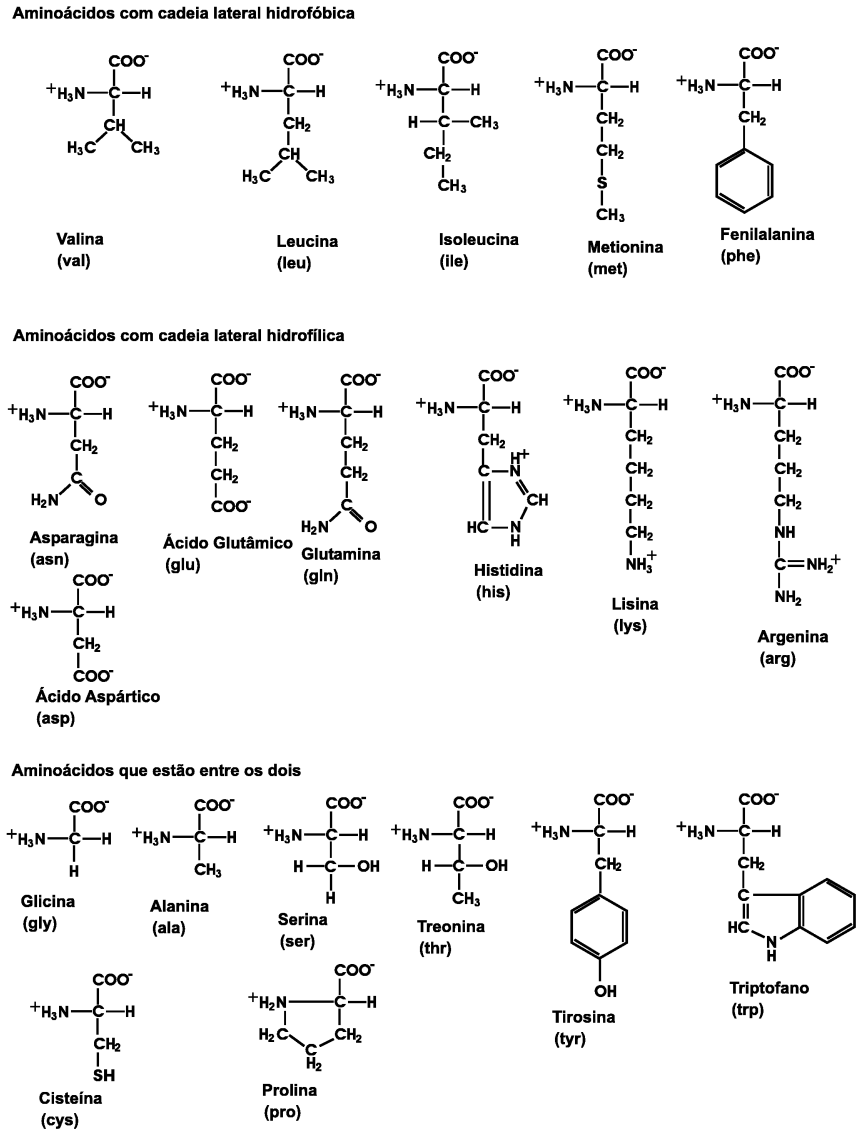


Figura 2.2: Classificação dos vinte aminoácidos padrões encontrados em proteínas.

Dependendo da natureza química da cadeia lateral, os aminoácidos podem ser divididos em três diferentes classes. A primeira classe compreende os aminoácidos com cadeia lateral estritamente apolares ou hidrofóbica (compostos que não se dissolvem na água, a saber: Alanina, Valina, Leucina, Isoleucina, Fenilalanina e Prolina). Aminoácidos que possuem cadeia lateral estritamente polares ou hidrofílica (compostos que se dissolvem em contato com a água) compõem a segunda classe, isto é: Ácido Aspártico, Ácido Glutâmico, Serina, Treonina, Cisteína, Asparagina, Glutamina, Histidina e Argenina. A terceira classe é composta pelos aminoácidos com características polares e apolares, sendo os mesmos também chamados anfipáticos, isto é: Lisina, Tirosina, Metionina, e Triptofano (PETSKO; RINGE, 2004).

2.2 Ligações Peptídicas e os Polipeptídeos

Os aminoácidos formam hétero-polímeros[1] (cadeias polipeptídicas) por meio de ligações covalentes denominadas ligações peptídicas, sendo que este processo de polimerização ocorre no ribossomo da célula (SCHULZ; SCHIRMER, 1979). Essas ligações ocorrem entre o grupo carboxila de um aminoácido e o grupo amina do outro (COPELAND, 1993). Durante o processo de ligação ocorre a perda de uma molécula de água. A Figura 2.3 representa o resultado de uma ligação peptídica.

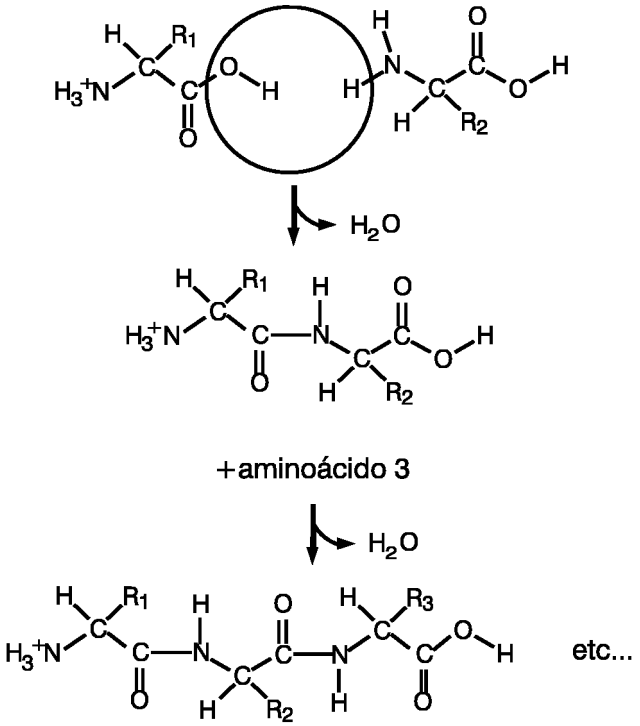


Figura 2.3: Processo de formação de uma ligação peptídica.

Quando vários aminoácidos estão conectados, o polímero resultante é denominado polipeptídeo. A diferença entre proteínas e polipeptídeos é basicamente semântica. Por definição, todas as proteínas são polipeptídeos, porém, costumam-se chamar de polipeptídeo apenas pequenas sequências de aminoácidos.

As ligações Peptídicas possuem algumas propriedades peculiares. A primeira delas é o fato do comprimento da ligação peptídica não poder ser medido como uma típica dupla ligação carboxílica (C=O) e uma ligação simples carbono-nitrogênio. Ambas as distâncias das ligações carboxílicas e carbono-nitrogênio estão nos valores intermediários, entre as distâncias conhecidas para compostos deste tipo já relatados. A explicação usual é que a dupla e simples ligação ficam alternando-se entre os pares OC e CN, isto é O=C-N e O-C=N (COPELAND, 1993). Observa-se que estas ligações ocorrem em uma estrutura planar; assim, os seguintes seis átomos C_i^α, C_i, O_i, N_{i+1}, H_{i+1}, C_{i+1}^α fazem parte de um

[1]Macromoléculas constituídas pela repetição de pequenas moléculas idênticas são ligadas covalentemente (LODISH et al., 2004).

mesmo plano, ou seja, carbonos α de aminoácidos adjacentes permanecem a um mesmo plano (SCHULZ; SCHIRMER, 1979).

Outra propriedade observada é que, embora a rotação sobre a ligação C-N seja restrita[2], rotações sobre o C_α-N e C-C_α podem ocorrer livremente, e são descritas por dois ângulos ϕ e ψ, associados respectivamente a cada uma das ligações (Figura 2.4) (RAMACHANDRAN; SASISKHARAN, 1968).

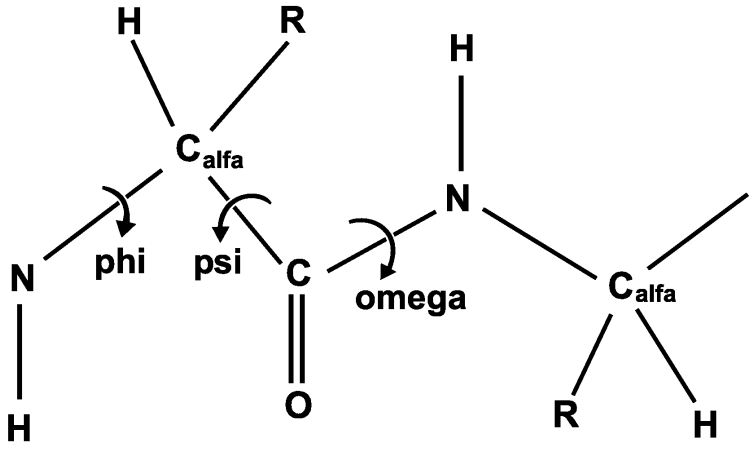


Figura 2.4: Ângulos ψ e ϕ.

A partir de experimentos laboratoriais, Ramachandran e seus colaboradores observaram que os pares ϕ e ψ concentram-se em duas regiões de valores específicos, conforme o mapa mostrado na Figura 2.5. Este fato é válido para todos os aminoácidos, exceto a *glicina*, que por apresentar uma cadeia lateral muito simples, possui menos restrições conformacionais. Observa-se também que cada uma das duas regiões de maior concentração pode estar associada a um tipo de estrutura secundária, a qual será discutida na Seção 2.4 (COPELAND, 1993). Os ângulos ϕ e ψ são chamados de ângulos diedrais e são responsáveis por definir a forma da cadeia principal do polipeptídeo. Os valores assumidos pelos ângulos diedrais respeitam a propriedade de que os carbonos α de dois aminoácidos adjacentes devem estar no mesmo plano. O ângulo ω é formado pela ligação covalente, parcialmente dupla, entre o Carbono (C) e o Nitrogênio (N), limitando seus valores, os quais podem assumir 180° graus ou 0° graus.

2.3 Estrutura Primária de Proteína

A sequência dos aminoácidos que compõem a proteína representa a estrutura primária da mesma. Com esta informação, pode-se apenas afirmar o número de resíduos e como estão ligados (Ligação Peptídica).

[2]onde C não é o carbono α.

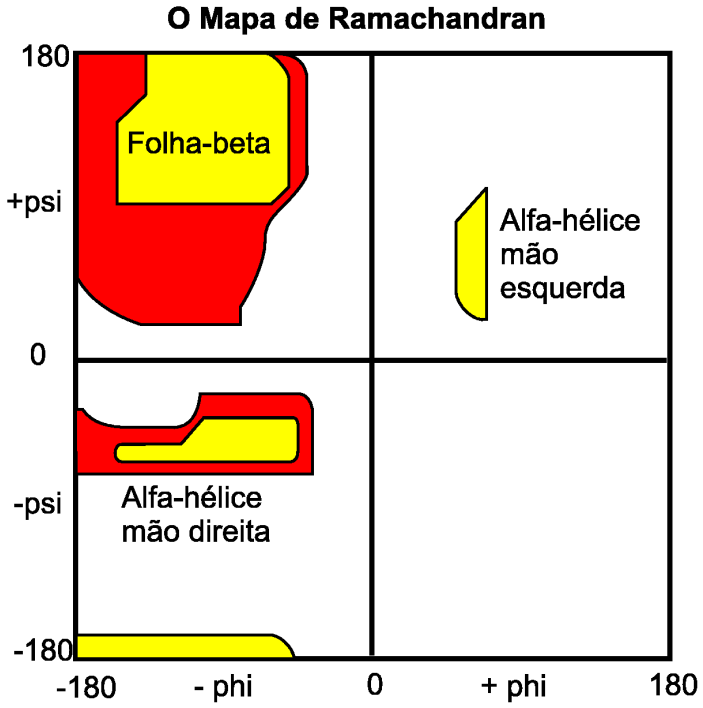


Figura 2.5: Mapa de Ramachandran.

A sequência de texto TTCCPSIVARSNFNVCRLPGTPEALCATYTGCIIIPGATCPGDYAN, foi obtida no *Protein Data Bank* (PDB)[3], constitui a estrutura primária da proteína 1CCN. Tendo tal informação e observando a Tabela 2.1, pode-se concluir que a proteína em questão é formada pelos aminoácidos: Treonina, Cisteína, Prolina, Serina, Isoleucina, Valina, Alanina, Asparagina, Arginina, Fenilalanina, Leucina, Glicina, Ácido Glutâmico e Tirosina, sendo que cada um deles aparece uma ou mais vezes, numa sequência específica, a qual identifica a proteína.

2.4 Estrutura Secundária de Proteína

Embora as proteínas sejam polímeros lineares, suas estruturas não são cordões aleatórios (LODISH et al., 2004). A grande parte das proteínas solúveis tem um centro empacotado, consistindo primariamente de aminoácidos hidrofóbicos. Esta observação pode ser explicada pela tendência que grupos hidrofóbicos possuem de evitar o contato com a água e de se agrupar. Outra característica interessante de cadeias polipeptídicas dobradas é que os segmentos da cadeia, em aproximadamente todas as proteínas, adotam conformações nas quais os ângulos de torção ϕ e ψ da cadeia principal se repetem em padrões regulares. Esses padrões regulares formam os elementos da estrutura secundária da proteína.

Definem-se, usualmente, dois tipos de elementos de estrutura secundária:

1. Hélice α,

[3]O PDB é uma das principais bases de dados de proteínas com estrutura terciária determinada por meio dos métodos experimentais.

2. Folhas β,

A Hélice α é uma estrutura em forma de bastão. Esta estrutura é estabilizada por pontes de hidrogênio entre os agrupamentos NH e CO da cadeia principal. Seu comprimento é de 12 resíduos, em média. Já o número de resíduos por volta é 3.6 (BRANDEN; TOOZE, 1991). A Figura 2.6 mostra uma representação desta estrutura.

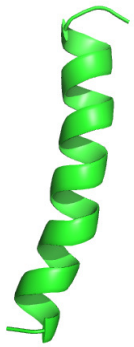


Figura 2.6: Estrutura Hélice α.

Já Folhas β apresentam uma cadeia principal distendida e, assim, não possibilita a existência de ligações de hidrogênio, o que libera o Oxigênio da carbonila e o Nitrogênio da cadeia principal para realizarem ligações de hidrogênio com partes distantes da cadeia principal. A Figura 2.7 representa a estrutura Folhas β, a qual é apresentada por flechas em razão de poder assumir direções na cadeia polipepitídica (BRANDEN; TOOZE, 1991).

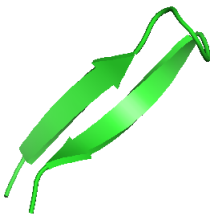


Figura 2.7: Estrutura Folhas β.

Tais tipos de estruturas secundárias são geralmente estabilizadas por ligações de hidrogênio, entre os grupos carboxila e amino do *backbone*. O *backbone* é também conhecido por cadeia principal da proteína.

2.5 Estrutura Terciária de Proteína

A estrutura terciária das proteínas refere-se à conformação total (arranjo tridimensional) de todos os resíduos de aminoácidos e é estabilizada, principalmente, pelo efeito hidrofóbico, pela ligação de hidrogênio entre as cadeias polares e pela força de van der Waals. A Figura 2.8 ilustra um exemplo da estrutura terciária de uma proteína.

Figura 2.8: Representação da estrutura terciária de uma proteína (PDB 1CCN).

A forma tridimensional assumida pela proteína é conhecida por estrutura nativa. Em sua estrutura nativa, as proteínas ocupam um estado de energia livre. Neste estado, os resíduos não polares estão afastados do meio aquoso, o assim conhecido centro hidrofóbico da proteína. De forma similar, no estado nativo, são favoráveis as interações entre os aminoácidos polares que se situam na superfície hidrofílica da proteína com o solvente. Proteínas em seu estado natural sempre se enovelam espontaneamente em suas respectivas estruturas tridimensionais (COPELAND, 1993).

A caracterização das estruturas terciárias é uma tarefa muito difícil. Uma das possibilidades é categorizada por meio de arranjos topológicos dos vários elementos da estrutura secundária. Uma característica da estrutura terciária é possuir uma superfície topográfica complexa que permite a proteína interagir especificamente com pequenas moléculas (que podem ligar-se em cavidades) e com outras macromoléculas, com as quais a proteína pode ter regiões de topologia complementar e de adequadas cargas (positiva ou negativa). Tais regiões são frequentemente formadas de extensões de aminoácidos unindo elementos de estrutura secundária (PETSKO; RINGE, 2004).

Conhecendo a estrutura terciária das proteínas é então possível analisar sua unidade fundamental conhecida como domínio. Entende-se, por domínio, uma parte da sequência da proteína e estrutura, a qual existe independentemente do restante da cadeia principal. Um domínio pode aparecer em uma variedade de proteínas evolutivamente relacionadas. O tamanho do domínio pode variar de 25 a 500 aminoácidos (LODISH et al., 2004).

Investigar e determinar as estruturas terciárias são muito importantes, visto que estrutura e função são fortemente correlacionadas.

2.6 Representação Computacional da Proteína

Nesta seção é discutido dois tipos de representações das proteínas: coordenadas internas e cartesianas.

As conformações de uma proteína são as estruturas tridimensionais que elas podem adotar. Desta forma, analisando as conformações é possível entender a sua influência nas propriedades da proteína. Diferentes conformações são conseguidas pelas mudanças no comprimento de ligação, nos valores dos ângulos de ligação e nos ângulos torsionais. Rotações em ligações simples produzem mudanças significativas na conformação. Por esta razão é muito comum que os algoritmos mantenham fixos tanto o comprimento de ligação como o valor dos ângulos de ligação, alterando-se somente os valores dos ângulos torsionais (LEACH; GILLET, 2007). A Figura 2.9 ilustra o ângulo torsional τ. Ele é definido como o ângulo entre os planos ABC e BCD.

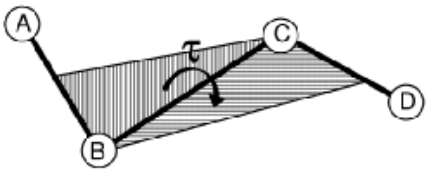


Figura 2.9: Representação do ângulo torsional τ.

Desta forma, modificando os valores dos ângulos torsionais, têm-se diferentes conformações, as quais podem ser representadas computacionalmente sob duas formas:

1. **Coordenadas internas**: A proteína é representada por meio de informações do comprimento de ligação entre dois átomos, o valor do ângulo de ligação com um terceiro átomo e, por fim, do ângulo diedral formado com um quarto átomo.

2. **Coordenadas Cartesianas**: A proteína é representada pela posição tridimensional de cada átomo que a constitui.

No trabalho de KOSLOVER; WALES (2007) foi realizada uma comparação da eficiência dos sistemas de coordenadas na otimização da geometria das proteínas. Foram empregadas as coordenadas Cartesianas e coordenadas internas em proteínas de tamanho variando de 16 a 999 resíduos. Além disso, o conjunto de teste possuía uma variada complexidade de hélices à beta-barril de enzima. A conclusão é uma dependência no tamanho da proteína, em que as coordenadas internas foram mais eficientes em proteínas pequenas. Já para outras proteínas as coordenadas Cartesianas apresentaram ser mais eficientes.

Vale ressaltar que no *framework* proposto a proteína é representada em ambos os sistemas.

2.7 O Problema do *Folding* de Proteínas

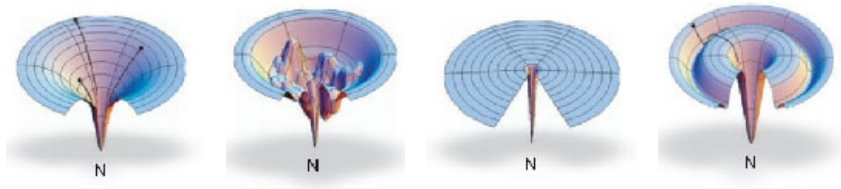
Possuindo a sequência de aminoácidos (estrutura primária) de uma proteína, qual seria sua estrutura nativa?

A resposta desta pergunta, aparentemente simples, ainda não se conhece, uma vez que o processo conhecido como *folding* de proteínas encontra-se sem solução, embora trabalhos pioneiros de relevância sobre o tema, como os de Christian Anfisen (ANFINSEN, 1973), sejam datados próximos do início da década de 70. Nestes trabalhos é enfatizado que somente a sequência de aminoácidos da proteína é necessário para determinar sua estrutura terciária. Um outro trabalho de relevância foi LEVINTHAL (1968), no qual se evidenciou que, embora há um montante elevado de conformações as quais uma proteína pode assumir durante o *folding*, a mesma deve empregar caminhos de *fold* peculiares que evitam extensivas buscas no espaço conformacional. Ou seja, um mecanismo aleatório que abrange todo o espaço conformacional não é apropriado, uma vez que não seria compatível com a escala de tempo biológica. Por conseguinte, no processo de *folding* existe um guia para as mudanças conformacionais da proteína.

Uma interpretação atual do processo termodinâmico do *folding* de uma proteína é destacada como sendo 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 (LEOPOLD; MONTAL; ONUCHIC, 1992) (ONUCHIC et al., 1996). Uma interpretação alternativa evoca o Controle Cinético, enfatizando-se que há um caminho de *folding* (*folding pathway*), ou seja, existem uma ou mais sequências de eventos sucessivos que levam a proteína sem padrão estrutural à estrutura nativa, que pode ou não ser a de menor energia livre.

No trabalho (PANDE; ROKHSAR, 1999) foi demonstrado, por meio de simulações empregando modelos simplificados, que uma proteína passa por vários caminhos intermediários até encontrar à sua estrutura nativa. Ou seja, existem vários caminhos de *folding* para uma mesma proteína. A Figura 2.10 ilustra uma representação de perfil enérgico do *folding* por meio do funil. Assim, podem-se visualizar, ao mesmo tempo, os caminhos de *folding*, intermediários, armadilhas cinéticas e até a velocidade do processo. Na Figura 2.10a é a representação de Proteína de *folding* rápido: não existem barreiras energéticas entre as conformações não-nativas e a nativa (*downhill folding*). A Figura 2.10b tem a representação de armadilhas cinéticas, pois espera-se encontrar caminhos intermediários fora do caminho de *folding*. Uma representação de *folding* lento está na Figura 2.10c, sendo que há grande número de conformações de mesma energia, a proteína passa muito tempo procurando aleatoriamente pelas mais estáveis. Por fim, por meio da Figura 2.10d há a representação de *folding* com um intermediário obrigatório. Como não há caminhos que não passem pelo mínimo local de energia, sempre haverá então pelo menos um intermediário (DILL et al., 2008).

O processo do *folding* de proteínas apresenta características físico-químicas, as quais



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

Figura 2.10: Representação do perfil energético de *folding* por meio do funil (DILL et al., 2008). O N representa a estrutura Nativa.

segundo KARPLUS; SHAKHNOVICH (1992) devem ser levadas em consideração. Tais características são descritas conforme se segue:

- rotação sobre a ligação C-N seja restrita[4], rotações sobre o C_α-N e C-C_α podem ocorrer livremente, sendo descritas pelos ângulos ϕ e ψ, associados respectivamente a cada uma das ligações (RAMACHANDRAN; SASISKHARAN, 1968);
- interações entre os aminoácidos, com as interações eletrostáticas;
- forças de dispersão (Van der Waals);
- restrições de volume;
- ligações de hidrogênio e pontes de dissulfeto;
- interações dos aminoácidos com o meio aquoso.

As características físico-químicas evidenciam, no que tange à estabilidade proteica, um equilíbrio extremamente delicado entre estruturas terciárias sem significado biológico e a estrutura nativa. Cada aminoácido da sequência primária faz muitos contatos, seja com o solvente ou com outros aminoácidos, e cada um deles contribui para a estabilidade proteica de alguma forma. Além dessas interações locais, há efeitos globais que são decisivos quando se trata dessa estabilidade. Tanto essas interações locais quanto esses efeitos globais serão discorridos com mais ênfase na Seção 2.8.

Na Seção 2.7.1 é descrito de modo sucinto acerca da determinação experimental da estrutura terciária de proteínas. Já na Seção 2.7.2 discute a respeito de sua predição.

2.7.1 Determinação Experimental da Estrutura Terciária de Proteínas

A estrutura terciária da proteína pode ser determinada experimentalmente por meio de dois métodos principais: cristalografia de raio-X e Ressonância Nuclear Magnética (RNM).

[4]onde C não é o carbono α.

No método de cristalografia de raio-X, a cristalização de proteínas nem sempre é possível garantir a geração de bons cristais, uma vez que exige experimentos com diferentes parâmetros, tais como pH, temperatura, concentração da proteína e a natureza do solvente. Logo, não é fácil predizer uma boa condição para que se torne possível a cristalização da proteína (DRENTH, 1994). As estruturas de raio-X são determinadas em diferentes níveis de resolução. Na resolução mais baixa, somente a forma da molécula é obtida, enquanto que na alta resolução a maioria das posições atômicas pode ser determinada com alto grau de exatidão. Na resolução intermediária, a dobra da cadeia polipeptídica é, geralmente, corretamente revelada, bem como as posições aproximadas das cadeias laterais, incluindo-se seus sítios ativos. A qualidade do modelo tridimensional final da proteína depende da resolução dos dados do raio-X e do grau de refinamento (BRANDEN; TOOZE, 1991).

Já no método de RNM, as propriedades de *spin* magnético do núcleo atômico da molécula são utilizadas para obter uma lista das restrições de distância entre os seus átomos, a partir da qual a sua estrutura tridimensional pode ser obtida. Este método não requer cristais de proteína e pode ser utilizado em moléculas proteicas em soluções concentradas. No entanto, sua utilização é restrita a pequenas moléculas de proteína (BRANDEN; TOOZE, 1991).

2.7.2 Predição de Estrutura Terciária de Proteína

Os métodos experimentais para determinação da estrutura terciária das proteínas possuem uma série de condições para que estes possam ser utilizados (Seção 2.7.1), o que torna bem relevante a investigação de métodos computacionais eficientes para a determinação da estrutura terciária, usualmente denominados de métodos para predição de estrutura terciária de proteínas. É estimado que menos de 1% das sequências primárias conhecidas tem-se a sua estrutura terciária (YANG; ZHANG, 2009).

Desde 1994, com o objetivo de mensurar a eficiência e a qualidade dos algoritmos para Predição da Estrutura da Proteína (PSP[5]), ocorre bianualmente o CASP (*Comparative Assessment of Methods for Protein Structure Prediction*) (MOULT et al., 2009). As sequências alvo são classificadas em duas categorias. Na categoria *template-based modelling*, os algoritmos podem empregar o uso de conhecimento das estruturas terciárias já conhecidas. Segundo KRZYSZTOF; GINALSKI (2006), esse método é dependente da acurácia do alinhamento, refinamento do modelo e da qualidade das estruturas conhecidas.

Por outro lado, em *template free modelling*, não é dependente de nenhum conhecimento das estruturas terciárias já conhecidas. Para tanto, este foca no uso dos modelos físicos que derivam somente das informações contidas na sequência primária alvo (ZHANG, 2008b).

Por ainda não haver uma teoria que descreva completamente o processo de *folding*, uma alternativa para o PSP tem sido as abordagens que a tratam como um problema de

[5]do inglês *Protein Structure Prediction*.

otimização. Na categoria *template-based modelling*, entre os métodos existentes, destacam-se aqueles baseados em comparativa ou por homologia (DOOLITTLE, 1986; HILBERT; BÖHM; JAENICKE, 1993) e *threading* (BAXEVANIS; OUELLETTE, 2001). Já para a categoria *template free modelling* são destacados os algoritimos *Ab initio* (CUI; CHEN; WONG, 1998; VULLO, 2002).

Nas subseções a seguir são explanados sucintamente os métodos de predições baseados em comparativa ou por homologia, *threading* e *Ab initio*.

2.7.3 Modelagem Comparativa ou por Homologia

Esta técnica de modelagem significa predizer a estrutura terciária de uma proteína desconhecida com base em uma estrutura conhecida de uma outra proteína, chamada de homóloga[6]. Conforme discorrido, dentre as técnicas de predição, esta é a técnica mais dependente dos dados experimentais. Por isso, não requer um alto esforço computacional (ECHENIQUE, 2007).

Para se conseguir uma boa acurácia de modelagem, é importante descobrir a quantidade de similaridade com a sequência conhecida, que é necessária para predizer a estrutura com exatidão. Para determinar essa similaridade, HILBERT; BÖHM; JAENICKE (1993) estudaram superposições de alinhamento de um vasto número de estruturas conhecidas, de diferentes formas e classes funcionais com diferentes graus de homologia. Com base neste estudo, Hilbert et al. sugeriram as seguintes relações entre sequências homólogas e diferenças estruturais:

- O tamanho do núcleo da região comum diminui conforme se reduz a identidade na sequência. Alinhamentos com mais de 50% de similaridade possuem acima de 90% de seus resíduos em regiões estruturalmente conservadas. Se a identidade na sequência fica abaixo de 20%, o núcleo da região comum contém cerca de 65% dos aminoácidos;

- Regiões estruturalmente divergentes, com mais de 50% de similaridade na sequência, possuem conformação estrutural parecida. Grandes desvios estruturais podem acontecer se a similaridade for baixa;

- A diminuição da correlação de similaridade na sequência implica em aumento no número de inserções e/ou remoções em uma das sequências para que se tornem iguais. Identificou-se que para um número máximo de 16 inserções e remoções, em geral, a similaridade é abaixo de 20%. Por outro lado, praticamente nenhuma inserção e remoção são verificadas com mais de 60% de similaridade.

Os estudos de Hilbert et al. não se esgotam o assunto de similaridade de proteínas. Assim, surgiram-se outros: KABSCH; SANDER (1983) demonstraram que até mesmo

[6]Proteínas homólogas possuem um ancestral comum.

uma similaridade exata, em pequenos segmentos, não fornece indicação de estrutura, apresentando exemplos de pentapeptídeos idênticos que participam de diferentes estruturas em diferentes proteínas. Em WILSON et al. (1985), estendeu-se essa ideia para hexapeptídeos. Entretanto, em COHEN; PRESNELL; COHEN (1993), examinando os hexapeptídeos conclui que, dentro de uma classe estrutural de proteína ou domínio, a similaridade na estrutura de um hexapeptídeo sequencialmente idêntico é preservada. Foi com esse estudo que se inspirou a possibilidade de desenvolver algoritmos para predizer as estruturas terciárias de proteínas com domínio conhecido (BARTON; COHEN; BRADFORD, 1993; PEITSCH, 2002).

Em suma, as técnicas de modelagem por homologia investigam um enovelamento desconhecido, modelando-o por intermédio das estruturas conhecidas. E no recente trabalho de KACZANOWSKI; ZIELENKIEWICZ (2010) destacou-se que proteínas homólogas quase sempre possuem estruturas terciárias semelhantes. Desta forma, a eficácia desse método é extremamente dependente da qualidade das estruturas das proteínas conhecidas.

2.7.4 Modelagem por *"Threading"*

As abordagens de *threading* e modelagem por homologia (Seção 2.7.3) são baseadas na observação de que muitas proteínas no PDB são muito similares. Sendo assim, muitos cientistas estão investigando que há somente um limitado número de *folds* de proteínas diferentes na natureza. As estimativas variam consideravelmente, mas prevêm-se que existam cerca de 1000 *folds* de proteínas. Isso propicia uma abordagem para a predição de estrutura terciária de proteínas, determinando a estrutura de uma nova proteína pela busca (*match*) de seu melhor ajuste para alguma estrutura particular na biblioteca de estruturas.

Embora as técnicas de predição *threading* e modelagem por homologia fundamentam-se com a similaridade das proteínas no PDB, para uma boa acurácia de tais técnicas, *threading* aplica-se quando a proteína não tem nenhuma homóloga, mas pode ter uma estrutura tridimensional similar. Já a modelagem por homologia, conforme já mencionado, necessita possuir uma proteína homóloga. Vale ressaltar que esta regra não é o único aspecto a ser analisado na decisão de qual das técnicas devem ser utilizadas.

O processo de determinação dos métodos de *threading* pode ser descrito da seguinte forma: obtém-se uma sequência de busca e tenta alinha-la em um modelo de estrutura escolhido aleatoriamente, a partir do conjunto de proteínas das principais estruturas tridimensionais já determinadas. O alinhamento da sequência de busca com o modelo de estrutura pode ocorrer das seguintes formas:

- Alinhamento sequência-sequência: busca-se encontrar o melhor alinhamento entre a sequência de busca e a sequência de aminoácidos do modelo de estrutura por meio

de inserções e remoções;

- Alinhamento sequência-estrutura: a sequência de busca é movimentada sobre a estrutura tridimensional sujeita à pré-determinadas restrições físicas, referentes ao tamanho dos elementos da estrutura secundária, às regiões de *loop* que podem ser fixas ou variáveis dentro de um intervalo, entre outras restrições.

Para cada posicionamento da sequência contra a estrutura, interações de pareamento e hidrofóbicas entre resíduos não locais são determinadas. Esses cálculos são usados para determinar o alinhamento mais favorável da sequência questionada contra o modelo de estrutura selecionado (BAXEVANIS; OUELLETTE, 2001). Análogo à modelagem comparativa, este método é dependente dos dados experimentais das estruturas terciárias conhecidas.

2.7.5 Modelagem *Ab initio*

Nas abordagens *Ab initio* ou por primeiros princípios, diferentemente da modelagem por homologia e *threading*, nenhuma similaridade na sequência é necessária em relação às proteínas de estrutura conhecida. O processo de determinação não depende de a proteína ter um *fold* similar conhecido. Este modelo está mais próximo do princípio físico e mais distante dos dados experimentais. Em virtude de não possuir uma larga gama de informação, este necessita de um esforço computacional elevado, para assim tentar suprir essa carência (ECHENIQUE, 2007).

As abordagens computacionais *Ab initio* típicas computam a estrutura tridimensional realizando buscas no espaço de conformações adequado (VULLO, 2002). Alguns modelos computacionais são baseados em métodos de otimização, o qual envolve dois aspectos: primeiro, a especificação da função de minimização e, segundo, a escolha do algoritmo de busca (KHIMASIA; COVENEY, 1997).

As funções de minimização são baseadas em leis físicas de movimentação em campos potenciais cuidadosamente planejados (dinâmicas moleculares) (VULLO, 2002). Na maioria dos casos, a função procura minimizar a energia livre da molécula, pois se sabe que a estrutura nativa das proteínas ocupa um estado que corresponde a um mínimo de energia do sistema (KHIMASIA; COVENEY, 1997).

O fato do espaço de busca crescer exponencialmente com o número de resíduos da proteína é um dos grandes desafios deste método. Em CUI; CHEN; WONG (1998) observou-se algumas outras informações referentes à estrutura de proteínas que podem ser utilizadas no processo de determinação de estrutura terciária, ou sejam:

1. Estruturas nativas de proteínas são compactas e têm um centro altamente enriquecido com resíduos hidrofóbicos;

2. A força hidrofóbica dirige o processo de *folding*; dificilmente resíduos não-polares são encontrados na superfície externa da proteína;

3. Proteínas globulares são organizadas hierarquicamente, isto é, estrutura secundária, estrutura terciária e estrutura quaternária[7];

4. As proteínas seguem caminhos de processo de *folding* evitando extensivas buscas no espaço conformacional (LEVINTHAL, 1968).

Em virtude do modelo *Ab initio* ser o empregado nesta tese e este se basear nos modelos físicos, a Seção 2.8 descreverá a respeito das interações da proteína.

2.8 Interações da Proteína

A fim de ser possível estudar as proteínas *in silico* é extremamente necessário entender as interações entre os seus resíduos, átomos ou grupos de átomos, além da interação das conformações tridimensionais com o meio. Muitas dessas interações são eletrostáticas por natureza, como por exemplo: interações entre os sítios carregados (as cadeias laterais da Arg, Lys, His, Glu, Asp, N e C-terminal da proteína, íons), dipolo (grupos: NH, NH_2, C=O, OH e água) e quadrupolo (cadeias laterais da Tyr, Phe, Trp) (SPOEL, 1996). Essas interações são modeladas em equações nas quais os seus parâmetros, oriundos de observações experimentais, são descritos nos campos de forças.

Conforme mencionado na Seção 2.7, existem os fatores locais e globais que contribuem para a estabilidade da proteína. Estes fatores serão descritos nas subseções de 2.8.1 a 2.8.5.

2.8.1 Entropia conformacional

Uma cadeia polipeptídica esticada, ou seja, sem qualquer padrão estrutural, possui muitos graus de liberdade, pois as únicas limitações são impostas pelas ligações peptídicas (ver Seção 2.2). O número de interações intra-cadeia e a relevância dos padrões estruturais aumentam progressivamente à medida que seu processo de *folding* avança. Ao final desse processo tem-se a estrutura nativa, a qual é bem organizada e com poucos graus de liberdade acessíveis. Por conseguinte, a entropia conformacional diminuiu uma vez que a cadeia ficou mais organizada (DILL; BROMBERG, 2002).

2.8.2 Proteína e o meio

Durante o processo de *folding* os grupos apolares e alguns polares e os ionizados formam um núcleo denso no interior da proteína, ao qual o solvente não tem acesso. Nesse núcleo

[7]Refere-se à relação espacial (ligadas por ligações não-covalentes) entre duas ou mais cadeias polipeptídicas para compor uma proteína.

existem, praticamente, somente as interações proteína-proteína. Nessa fase há um grande custo energético, já que esse núcleo vai para um ambiente não aquoso e os grupos carregados e os polares interagem intensamente com a água. Porém, esse custo é compensado pela termodinâmica das moléculas de águas, as quais estavam presas à proteína nas camadas de solvatação sendo que, agora estão livres no solvente e, então, um aumento da entropia do sistema é observado (DILL; BROMBERG, 2002).

2.8.3 Empacotamento

As proteínas são tão densas quanto cristal orgânico; assim, elas devem ter empacotamento muito eficiente. Entenda-se por empacotamento, a capacidade de uma proteína explorar ao máximo seus contatos com ela mesma. Ou seja, uma proteína, a qual contem 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.

SEELIGER; GROOT (2007) demonstraram que o alto grau de empacotamento é uma propriedade universal das proteínas. Esta característica foi observada em qualquer que seja seu tamanho, função ou estrutura, pois a proteína apresentará esta propriedade, o que resulta em alta densidade.

O empacotamento contribui na estrutura proteica, uma vez que a sua forma nativa é bem empacotada. Assim, alguns graus de liberdade da cadeia não podem ser acessados, refletindo-se na estabilidade da estrutura.

2.8.4 O efeito hidrofóbico

O efeito hidrofóbico tem a finalidade de descrever a termodinâmica do comportamento de substâncias apolares em um solvente polar. Esse efeito é considerado o mais importante no processo do *folding* de proteínas, pois o mesmo não depende de interações específicas, além de que está sempre presente (DILL; BROMBERG, 2002).

No trabalho de LI; TANG; WINGREEN (1997) foi realizado um estudo estatístico acerca da importância desse efeito no processo de *folding*. Uma constatação matemática de que o efeito hidrofóbico origina a principal força indutora do processo de *folding* foi o resultado obtido. Já em DILL et al. (2008) comprovou-se que simplesmente classificar os aminoácidos em hidrofóbico e hidrofílico é o suficiente para prever, com sucesso, uma estrutura nativa de proteínas pequenas composta por hélices.

A hidrofobicidade mostra que a proteína, em sua estrutura nativa, apresenta duas regiões distintas: Uma superfície exposta ao solvente e com grande maioria dos resíduos polares, e uma outra com um núcleo constituído por uma grande maioria dos resíduos apolares (LEACH, 2001).

A estabilidade da proteína é extremamente influenciada por esse efeito. Pode-se dizer que dois fatos ilustram essa influência:

1. Os resíduos envolvidos na formação do núcleo são conservados pela evolução. Por sua vez, as mutações nesses resíduos têm uma grande probabilidade de resultar em perda de estabilidade da proteína.

2. A capacidade de uma proteína, a qual foi realizada a sua desnaturação, depende somente dos resíduos do núcleo hidrofóbico para retornar à sua estrutura nativa.

2.8.5 Forças indutoras do processo de *folding*

As forças indutoras do processo de *folding* são todas de natureza eletromagnética. Tais forças atuam somente em partículas as quais possuem carga elétrica. A sua interação tem alcance infinito, porém, com intensidade intermediária. Uma outra característica dessas forças tange na classificação das mesmas em covalentes e não-covalentes (DILL; BROMBERG, 2002).

Nas próximas seções serão descritas as principais forças, a saber: dispersão de London, potencial de Van der Waals, pontes salinas, ligações de hidrogênio e pontes de dissulfeto.

Forças de dispersão de London

Com a mecânica quântica foi possível comprovar que os elétrons não se movem ao redor do núcleo em órbitas bem definidas, mas sim de forma probabilística. Ou seja, há uma grande probabilidade de que a densidade eletrônica não seja uniforme e simétrica nos orbitais, sendo então possível o surgimento de polaridade temporária ou instantânea em uma molécula. Essa polaridade induz dipolos em moléculas vizinhas de forma que a interação resultante faz uma força atrativa fraca: a força de dispersão de London (LEACH, 2001).

A força de London é um dos tipos das forças de Van der Waals, que em geral englobam todas as interações não-covalentes entre moléculas não-iônicas. Uma aproximação frequentemente empregada para descrever o comportamento desta força é o potencial de Lennard-Jones, também conhecido como potencial L-J ou potencial $6 - 12$. Esse potencial considera, além da força de London, uma força repulsiva a qual surge quando átomos se aproximam muito. Essa força repulsiva pode ser entendida como uma consequência da combinação do princípio da exclusão de Pauli e da repulsão elétrica entre as nuvens eletrônicas (DILL; BROMBERG, 2002).

As interações são calculadas sobre pares de átomos. Em princípio, todas as interações de todos os pares de átomos deveriam ser avaliadas, mas isto elevaria significativamente o custo computacional. Sendo assim, define-se previamente a máxima distância que evidenciará uma interação de van der Waals. Outro valor de corte estabelecido é quando a

distância entre os átomos se torna menor que uma distância pré-definida, conhecida como corte de diminuição, pois, neste caso, $V(r) \to \infty$.

Existe uma relação entre a força de London e o efeito hidrofóbico no qual a torna a principal força indutora do processo de *folding*. Assim, quanto maior o empacotamento devido ao efeito hidrofóbico (Ver Seção 2.8.4), mais contatos entre cadeias laterais apolares se formam e as interações de London se estabelecem. Embora sua intensidade seja muito baixa, essa interação é a mais numerosa na estrutura nativa. Logo, a sua contrubuição para a estabilidade proteica é notória (DILL; BROMBERG, 2002).

Potencial de Van der Waals

O potencial de Van der Waals é utilizado para estimar a parte das energias de interações entre os grupos do tipo apolar-polar e polar-polar. Este potencial é baseado na força de dispersão de London, na repulsão entre as nuvens eletrônicas de grupos vizinhos e na interação eletrostática.

Pontes Salinas

As Pontes Salinas originam-se da interação entre uma cadeia lateral carregada positivamente e uma carregada negativamente. Ou seja, elas tratam de uma interação entre íons que pode ser equacionada pela lei de Coulomb, onde q_i e q_j são as cargas efetivas, r a distância entre elas e o termo $\frac{1}{4\pi\epsilon_o}$ é uma constante de proporcionalidade envolvendo a permissividade elétrica do meio (ϵ_o)

As pontes salinas mais importantes formam-se nos núcleos hidrofóbicos, quando dois ou mais grupos ionizados estão dentro do núcleo criam um ambiente essencialmente apolar. O custo energético envolvido nesse processo é muito elevado em virtude da interação favorável com a água dos grupos carregados que estão expostos ao solvente. Mesmo não sendo a melhor opção do ponto de vista energético, elas são importantes para a especificidade de uma conformação, pois desestabilizam aquelas nas quais a interação entre os íons não é ótima (DILL; BROMBERG, 2002).

Ligações de Hidrogênio

A Ligação de Hidrogênio é um tipo especial de interação entre os grupos polares. Ela surge quando o grupo doador possui um átomo de hidrogênio ligado à um elemento muito eletronegativo (como flúor, oxigênio ou nitrogênio) e o receptor possui um átomo muito eletronegativo.

No grupo do doador, a grande eletronegatividade do átomo o qual está ligado ao átomo de hidrogênio causa um grande deslocamento de sua nuvem eletrônica e, assim, deixa seu único próton praticamente exposto. Então, a carga parcial positiva do hidrogênio do doador interage com a carga parcial negativa do receptor. Desse fenômeno surge a Ligação

de Hidrogênio. O referido deslocamento, além de gerar as cargas parciais, permite uma maior aproximação dos grupos doador e receptor, uma vez que diminui o efeito da repulsão das nuvens eletrônicas. Essa menor distância tem dois efeitos (HONIG, 1999):

1. Intensificação da interação;

2. Aproximação das cadeias laterais próximas aos grupos, em que se aumenta o nível de empacotamento local.

As Ligações de Hidrogênio são importantes para o processo de *folding*, assim como as pontes salinas, as quais contribuem para a especificidade da estrutura nativa em virtude do efeito do empacotamento. Entretanto, a sua contribuição energética para a estabilidade é muito pequena ou nula.

Ponte Dissulfeto

A Ponte dissulfeto, também conhecidas como ligações S-S, é uma ligação entre grupos tiol (-SH) de cadeias laterais de dois resíduos de Cisteína (Cys), os quais podem ser da mesma cadeia polipeptídica ou de cadeias diferentes.

Em proteínas de cadeia única, a ponte de dissulfeto estabiliza a estrutura, pois mantém um contato nativo por meio de uma ligação covalente e por favorecer a formação de mais desses contatos nos resíduos vizinhos. Já nas proteínas com mais de uma cadeia peptídica, as pontes conferem uma interação forte o suficiente para mantê-las unidas mesmo quando a proteína é enviada para um outro meio.

A Ponte dissulfeto é a única interação covalente que influencia o processo de *folding*. As ligações peptídicas não são incluídas nessa classe, pois quando é comparada a estrutura nativa com uma estrutura esticada da proteína, não há diferença entre as ligações peptídicas. Porém, as pontes dissulfeto podem estar ausentes ou organizadas de forma diferente em uma estrutura não-nativa.

2.9 Considerações Parciais

Neste capítulo foi explanado os principais aspectos envolvendo as protéinas e elucidado os desafios, devido sua complexidade para o problema de predição de estruturas terciárias de proteínas *in silicio*. Podem-se enfatizar alguns aspectos sobre as proteínas, ou sejam:

- As proteínas são compostos orgânicos, constituído por compostos mais simples, denominados aminoácidos, os quais possuem um carbono central, C_α, que possui quatro ligantes diferentes: um grupo amino, um grupo carboxila, um hidrogênio e um radical ou cadeia lateral. Os aminoácidos são diferenciados por mudanças no radical. Pequenas sequências de aminoácidos são chamados polipeptídeos;

- As proteínas são moléculas hierarquicamente estruturadas, ou seja, possuem uma estrutura primária (seqência linear dos aminoácidos), estrutura secundária (conformações locais repetidas em quase todas as proteínas) e a estrutura terciária (arranjo tridimensional da molécula proteica).

- A relevância em investigar a estrutura terciária da proteína está em virtude de ser possível determinar qual a função da proteína no organismo. Este conhecimento contribui para o desenvolvimento de novos fármacos, pois conhecendo a estrutura tridimensional, torna-se então possível determinar quais os melhores compostos podem ligar-se ao sítio ativo da proteína.

Um destaque é a descrição sobre o problema de *folding* e as interações da proteína para a realização deste processo. Destas interações é que serão baseados os objetivos para a exploração do espaço de busca.

Capítulo 3

Computação Evolutiva

A abordagem Evolutiva ou Computação Evolutiva (CE) trata de inspirar-se nos fenômenos que ocorrem na natureza para solucionar problemas, principalmente na área de otimização. Uma das motivações de aplicar CE justifica-se pela sua capacidade de lidar com problemas para os quais não é possível, ou é difícil, obter uma descrição detalhada dos mesmos, ou ainda, não se consegue impor restrições rígidas ao escopo do problema de otimização. Uma outra motivação é capacitar o computador a tomar decisões, antes restritas a especialistas humanos, possuindo como informação somente as consequências das ações tomadas anteriormente, constituindo-se assim a evolução do processo de aprendizagem (MICHALEWICZ, 1996).

Neste capítulo, a Seção 3.1 trata da otimização multi-objetivo no que diz respeito ao conceito e definições. A Seção 3.2 referencia sobre a base fundamental dos algoritmos genéticos (Seção 3.3), ou seja, a teoria da evolução. A Seção 3.4 refere-se aos algoritmos evolutivos aplicados nos problemas de otimização multi-objetivo. Já as Subseções 3.4.1 e 3.4.2 tratam os algoritmos evolutivos multi-objetivo empregados.

3.1 Otimização Multi-Objetivo

Os Problemas de Otimização Multi-Objetivo (POMO) são utilizados quando a quantidade de objetivos não seja único e que há a necessidade de serem tratados simultaneamente. Além disso, tais objetivos podem ser conflitantes entre si. Neste tipo de problema, há um grupo de soluções que satisfaz um "equilíbrio" de situações (soluções) (Coello Coello, 2006).

Um POMO possui um conjunto de funções objetivos a serem otimizadas e restrições que devem ser satisfeitas por qualquer solução factível[1] (DEB, 2001). O conjunto de todas as soluções factíveis é conhecido como *região factível*.

[1]Uma solução x é factível se, e somente se, satisfazer todas as restrições. Caso contrário, a solução será não factível.

Para os algoritmos de otimização, todas as funções objetivos devem ser maximizadas ou minimizadas.

O *espaço de objetivos* **Z** é um espaço multi-dimensional, composto pelo vetor funções objetivos $f(x)$. A diferença entre multi-objetivo e mono-objetivo é o espaço de busca: no multi-objetivo é multidimensional, i. e., cada solução x, no espaço de decisão, possui $f(x)$ em **Z**; e no mono-objetivo é unidimensional.

3.1.1 Soluções de Pareto Ótimas

Apresentar uma decisão implica em considerar vários aspectos visando encontrar a melhor solução. Pode haver situações que, considerando somente as características quantitativas, não se consegue determinar uma solução melhor que a outra. Toma-se como exemplo aquele apresentado em TICONA (2003): para a decisão da compra de um carro, pode-se considerar que se está procurando o carro com melhores preço e conforto. A Figura 3.1 ilustra essas opções.

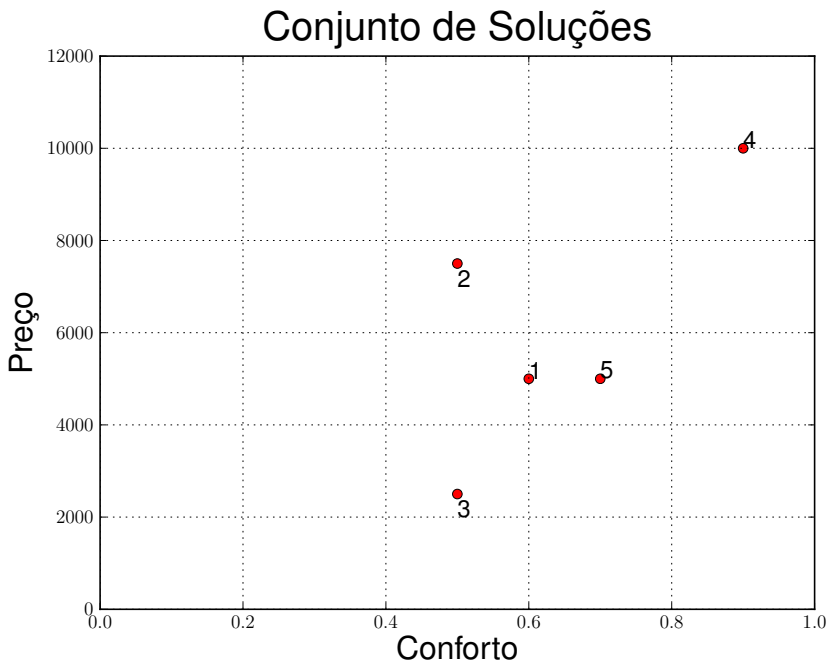


Figura 3.1: Exemplo do multi-objetivo (TICONA, 2003).

O objetivo é minimizar preço e maximizar conforto. Neste caso, tem-se cinco possíveis alternativas de compra. As soluções 1 e 2, são descartadas, pois a solução 5 fornece mais conforto por um igual preço e preço inferior, respectivamente. As soluções 3, 4 e 5 são as melhores alternativas de compra, mas, em termos quantitativos, não se pode afirmar quem é a melhor. Pode-se atribuir um "compromisso" entre os objetivos. Quanto maior o conforto, maior o preço e vice-versa (TICONA, 2003).

Uma solução *domina* uma outra solução se, e somente se, em todos seus objetivos, possuir valores melhores. No exemplo de TICONA (2003), a solução 5 domina a solução 1 e *não é dominada* por nenhuma outra. Aplica-se o mesmo raciocínio para as soluções 3 e 4.

Não tendo mais informações a respeito das soluções, pode-se afirmar que o conjunto das soluções 3, 4 e 5 são aquelas melhores, o qual é também conhecido como *conjunto não dominado*. Logo, as soluções 1 e 2 constituem o *conjunto dominado*.

O conjunto das soluções não dominadas pode ser representado no espaço cartesiano e formam a chamada *frente de Pareto* [2] ou *fronteira de Pareto*. As *soluções Pareto ótimas* ou conjunto Pareto ótimo, ou ainda, fronteira ótima de Pareto, formam o conjunto de soluções não dominadas em relação a todas as soluções possíveis.

3.1.2 Metas em Otimização Multi-Objetivo

Em DEB (2001) é assinalada duas importantes metas em otimização multi-objetivo:

1. Encontrar um conjunto de soluções que esteja o mais próximo possível do conjunto Pareto ótimo;

2. Encontrar um conjunto de soluções com maior diversidade possível.

A primeira meta é comum para todos os problemas de otimização, pois soluções muito distantes da fronteira ótima de Pareto são indesejáveis. No entanto, a segunda meta, encontrar a maior diversidade, é uma tarefa específica para a otimização multi-objetivo. Em POMO, trabalha-se com o espaço de decisões e o espaço de objetivos, sendo imprescindível que as soluções tenham uma boa diversidade nestes espaços.

3.2 Teoria da Evolução

A teoria da evolução foi proposta por Charles Darwin (DARWIN, 1859) na década de 1850 e até nos dias atuais é o principal conceito unificador das diversas áreas da biologia. Tal teoria começou a ser desenvolvida a partir das observações de Darwin durante sua viagem a bordo do navio *Beagle*. Esta teoria tem como um de seus princípios o conceito de seleção natural, o qual afirma que o meio atua sobre os indivíduos, selecionando os mais adaptados ao ambiente para sobreviver, pois as populações não podem crescer demais. São considerados indivíduos adaptados ao ambiente aqueles que conseguem sobreviver e deixar descendentes.

Darwin não conseguia explicar, geneticamente, como a variabilidade dos indivíduos surgia e era transmitida para os descendentes. Só em 1900, nos estudos de Gregor Mendel,

[2] Vilfredo Pareto, economista e sociólogo italiano. Graduou-se na universidade de Turin em 1869 e trabalhou como engenheiro em uma grande companhia ferroviária. Em 1893 foi lecionar na universidade de *Lausanne*, França (BRITANNICA, 2007).

torna-se possível explicar a ligação entre os mecanismos de herança e o cromossomo, dando-se origem à genética (JUNIOR; SASSON, 2003).

Em 1940, pesquisadores com o auxílio da teoria genética chegaram à Teoria Sintética da Evolução ou Neodarwinismo (JUNIOR; SASSON, 2003), baseada nos conceitos de recombinação gênica e mutação. A recombinação gênica é responsável pela transmissão das características dos pais para os filhos. A mutação é responsável pelo surgimento da diversidade nos indivíduos da população, com o surgimento de novas características que, se forem benéficas, tornam os indivíduos mais aptos e adaptados, facilitando-se a geração de descendentes com tais características; caso contrário, essas características tendem a ser eliminadas. Esse processo é denominado de seleção natural.

3.3 Algoritmos Genéticos

HOLLAND (1975) introduziu os Algoritmos Genéticos (AGs) com a motivação de estudar, formalmente, os conceitos de adaptação que ocorrem na natureza, formalizá-los matematicamente e desenvolver sistemas artificiais[3] que imitassem os mecanismos originais encontrados em sistemas naturais.

O AG proposto por Holland é um método que consiste em modificar uma população[4] inicial em uma nova população utilizando a seleção natural e os operadores genéticos: recombinação gênica (ou *crossover*) e mutação. Os AGs utilizam uma terminologia originada da teoria da evolução natural (Seção 3.2) e da genética. Um indivíduo da população é representado por um único cromossomo, que contém a codificação (genótipo) de uma possível solução do problema (fenótipo). Cromossomos são geralmente implementados na forma de listas de atributos, vetores ou *strings*, onde cada atributo é conhecido como gene e os possíveis valores que um determinado gene pode assumir são denominados alelos. No AG proposto por Holland, um cromossomo é geralmente representado por uma *string* binária, ou seja, uma *string* de zeros e uns.

Segundo MICHALEWICZ (1996), um AG busca um espaço de soluções potenciais para o problema e para isso requer um equilíbrio entre dois objetivos aparentemente conflitantes: o aproveitamento das melhores soluções e a exploração do espaço de busca. AGs constituem, assim, uma classe de métodos de busca de propósito geral que apresentam um balanço considerável entre aproveitamento de melhores soluções e exploração do espaço de busca.

Mesmo apresentando etapas não-determinísticas em seu desenvolvimento, os AGs não são métodos de busca puramente aleatórios, isto em consequência à combinação de variações aleatórias com seleção pelos valores de adequação (*fitness*) atribuído a cada indivíduo.

[3]Simulados em computador.

[4]Conjunto de indivíduos representando as soluções candidatas codificadas de forma similar a cromossomos em genética.

Uma propriedade importante dos AGs é que esses mantêm uma população de soluções candidatas enquanto que outros métodos alternativos, como *simulated annealing* (AARTS; KORST, 1989), analisam um único ponto no espaço de busca a cada instante. Além disso, os AGs possuem um paralelismo implícito decorrente da avaliação independente de cada uma das cadeias de *bits* (cromossomo) que compõem os indivíduos. O processo de busca é multi-direcional, com a manutenção de soluções candidatas que representam a busca em várias partes do domínio e com troca de informações entre essas soluções. A cada geração, soluções relativamente "boas" geram mais descendentes, enquanto que soluções relativamente "ruins" tendem a ser eliminadas. Para fazer a distinção entre diferentes soluções, é empregada a função de avaliação (*fitness*) que simula o papel da pressão exercida pelo ambiente sobre o indivíduo.

Para desenvolver um AG, em um problema particular, deve-se considerar os seguintes componentes:

- ❑ Representação genética para soluções potenciais (etapa de codificação);
- ❑ Procedimento para criar uma população inicial;
- ❑ Função de avaliação para classificar as soluções em termos de sua adaptação ao ambiente (sua capacidade de resolver o problema);
- ❑ Definir os operadores genéticos[5] com base na codificação (representação dos dados referentes ao indivíduo) utilizada;
- ❑ Valores para os diversos parâmetros do AG, tais como: tamanho da população e probabilidades de aplicação dos operadores genéticos.

3.3.1 Codificação dos Indivíduos

A codificação é uma das etapas mais críticas na definição de um AG. No AG clássico[6] os indivíduos da população são codificados em *strings* binárias de tamanho fixo. A grande motivação para o emprego da codificação binária está na Teoria de Esquemas (HOLLAND, 1992) utilizada para justificar a eficiência dos AGs, sendo concluído que a representação binária maximiza o paralelismo implícito inerente ao AG. Entretanto, tanto MICHALEWICZ (1996) como DEB (2001) apresentam resultados de comparações do desempenho de AGs com codificação binária e com ponto flutuante. Os resultados apresentados revelam superioridade da codificação em ponto flutuante quando comparada com a codificação binária.

MICHALEWICZ (1996) argumenta que a representação binária não é adequada quando o espaço de busca é de alta dimensão. Porém, esta argumentação não é muito aceita na

[5] Ver na Seção 3.3.3.
[6] Proposto por Holland.

literatura sobre AGs. Espaços de busca de alta dimensão podem às vezes ser explorados eficientemente, enquanto que espaços de busca de dimensão reduzida podem apresentar dificuldades significativas. Outro problema encontrado com a codificação binária ocorre quando o espaço de busca do problema é contínuo, podendo ocorrer Hamming *cliffs* com certas *strings*, por exemplo 01111 e 10000, onde a transição para uma solução vizinha no espaço de números de ponto flutuante requer a alteração de muitos *bits* da *string* (DEB, 2001). Os Hamming *cliffs* presentes na codificação binária causam o atraso para uma busca gradual nos espaços de busca contínuos.

Outra dificuldade no caso de problemas com espaços de busca contínuos é a incapacidade de armazenar qualquer precisão arbitrária na solução ótima, sendo isto necessário quando a codificação binária é utilizada para escolher a priori o tamanho da *string* para que o AG seja capaz de armazenar uma certa precisão na solução. Quanto mais precisão for requerida, então maior será o tamanho da *string*. Para grandes *strings*, requer-se então uma população grande, aumentando assim a complexibilidade do algoritmo, tornando-o então inviável (DEB, 2001). DEB (2001) também apresenta um operador de *crossover* para AGs com codificação de ponto flutuante que simula o princípio do operador de *crossover* de um ponto para AGs utilizando a codificação binária.

A definição inadequada da codificação pode acarretar problemas de convergência prematura[7] do AG. A estrutura de um cromossomo deve representar uma solução como um todo e deve ser a mais simples possível.

Em MICHALEWICZ (1996) são referenciados que, nos problemas de otimização com restrição, há a possibilidade de que os indivíduos modificados por *crossover*/mutação sejam inválidos. Nesses casos, cuidados especiais devem ser tomados na definição da codificação e/ou dos operadores.

3.3.2 Definição da População Inicial

Quando não há algum conhecimento do problema, o método para inicializar a população é aleatório. Deve atentar-se para os problemas com restrições visando não gerar indivíduos inválidos na etapa de inicialização. Conforme mencionado em LIMA (2006), no caso de codificação binária, se é sabido que a solução final vai apresentar mais $0'$s do que $1'$s, então esta informação pode ser utilizada, mesmo que não se saiba exatamente a proporção.

3.3.3 Operadores Genéticos

Os operadores genéticos mais frequentemente utilizados em AGs são o *crossover* e a mutação.

[7] A convergência prematura ocorre quando indivíduos relativamente adaptados, contudo não ótimos, rapidamente dominam a população, fazendo-se com que o AG convirja para um máximo ou mínimo local. Este problema pode ocorrer devido a uma formulação inadequada do problema.

Operador de *crossover*

O operador de *crossover* ou recombinação cria novos indivíduos utilizando a combinação de dois ou mais indivíduos. Estes indivíduos são chamados pais. No operador de *crossover*, há a troca de informação entre diferentes soluções candidatas. No AG clássico é atribuída uma probabilidade fixa de ocorrer *crossover* aos indivíduos da população.

O tipo de *crossover* mais difundido é aquele de um ponto. Para a aplicação deste, são selecionados dois indivíduos (pais) e, a partir de seus cromossomos, são gerados dois novos indivíduos (filhos). Para gerar os filhos, seleciona-se um mesmo ponto de corte aleatoriamente nos cromossomos dos pais; então, os segmentos de cromossomo criados a partir do ponto de corte são trocados.

Muitos outros tipos de *crossover* têm sido propostos na literatura. Alguns, exclusivos, quando utiliza-se codificação em ponto flutuante. Um exemplo é o *crossover* de mistura (BLX-α) (ESHELMAN; SCHAFFER, 1993). Considere x^1 e x^2 dois indivíduos selecionados para *crossover* e assume-se que $x_i^1 < x_i^2$, onde i representa o i-ésimo gene. O BLX-α escolhe aleatoriamente uma solução no intervalo $[x_i^1 - \alpha(x_i^2 - x_i^1), x_i^2 + \alpha(x_i^2 - x_i^1)]$. A literatura tem reportado que o melhor valor para α é 0.5 sobre qualquer outro valor escolhido. Se a diferença entre os pais for pequena, então a diferença entre os pais e os filhos também será pequena e vice-versa. Esta propriedade permite que este operador execute uma busca pelo espaço inteiro, no início, e também execute uma busca localizada quando a população tende a convergir para uma região do espaço de busca.

Operador de Mutação

O operador de mutação altera aleatoriamente um ou mais genes de um cromossomo. A taxa de mutação é a probabilidade de ocorrência de mutação em um gene. A finalidade do operador de mutação é criar uma variabilidade extra na população, mas sem destruir o progresso já obtido com a busca.

Para elucidar, toma-se a exemplo, a codificação binária. O operador de mutação padrão simplesmente troca o valor de um gene em um cromossomo (HOLLAND, 1992). Assim, se um gene selecionado para mutação tem valor um, o seu valor passará a ser zero após a aplicação da mutação, e vice-versa.

Segundo MICHALEWICZ; SCHOENAUER (1996), nos problemas com codificação em ponto flutuante, os operadores de mutação mais populares são a mutação uniforme e a mutação gaussiana. O operador para mutação uniforme seleciona aleatoriamente um componente $k \in \{1, 2, ..., n\}$ do cromossomo $x = [x_1, ..., x_k, ..., x_n]$ e gera um indivíduo $x' = [x_1, ..., x'_k, ..., x_n]$, onde x'_k é um número aleatório (com distribuição de probabilidade uniforme) amostrado no intervalo $[LB, UB]$, onde LB e UB são, respectivamente, os limites inferior e superior para o valor do alelo x_k. No caso da mutação gaussiana, todos os componentes de um cromossomo $x = [x_1, ..., x_k, ..., x_n]$ são modificados da seguinte

forma:

$$x' = x + N(0, \sigma),$$

onde $N(0, \sigma)$ é um vetor de variáveis aleatórias gaussianas independentes, com média zero e desvio padrão σ. Outro operador de mutação, especialmente desenvolvido para problemas de otimização com restrições e codificação em ponto flutuante, é a chamada mutação não-uniforme, destinada a realizar pequenos ajustes necessários para atingir a solução ótima junto aos indivíduos da população. Este e outros exemplos de operadores de mutação para problemas de otimização numérica podem ser encontrados em MICHALEWICZ (1996) e em MICHALEWICZ; SCHOENAUER (1996).

3.3.4 Seleção dos Indivíduos

O AG proposto por Holland utiliza um método de seleção de indivíduos para a próxima geração chamado técnica da roleta (MICHALEWICZ, 1996). A técnica da roleta atribui a cada indivíduo de uma população uma probabilidade de passar para a próxima geração que é proporcional ao *fitness* do indivíduo e à somatória do *fitness* de todos os indivíduos da população. Assim, quanto maior o *fitness* de um indivíduo, maior a probabilidade deste passar para a próxima geração. Sendo assim, a seleção de indivíduos pela técnica da roleta pode fazer com que o melhor indivíduo da população seja perdido, ou seja, não passe para a próxima geração. Uma alternativa é escolher como solução o melhor indivíduo encontrado em todas as gerações do algoritmo. Pode-se, também, manter sempre o melhor indivíduo da geração atual na geração seguinte, estratégia essa conhecida como seleção elitista (FOGEL, 1994; MICHALEWICZ, 1996).

A literatura relata outros mecanismos de seleção, sendo que dentre essas destacam-se a baseada em *rank* (BACK; FOGEL; MICHALEWICZ, 1997) e seleção por Torneio. A primeira estratégia utiliza as posições dos indivíduos ordenados de acordo com o *fitness* para determinar a probabilidade de seleção. Podem ser usados mapeamentos lineares ou não-lineares para determinar a probabilidade de seleção. Já a segunda, um número m de indivíduos da população é escolhido aleatoriamente para formar uma sub-população temporária. Deste grupo, o melhor indivíduo é selecionado. Assim, escolhe-se cada indivíduo que irá compor o grupo de N indivíduos selecionados.

Os mecanismos de seleção têm sido também empregados para determinar aqueles indivíduos que irão sofrer *crossover* e mutação. O número de indivíduos selecionados para *crossover* pode ser bem menor que o total de indivíduos da população, indicando que só alguns terão maior probabilidade de gerar descendentes em grande número.

3.4 Algoritmo Evolutivo Multi-objetivo

Os Algoritmos Evolutivos Multi-Objetivo (MOEA, do inglês *Multi-Objetive Evolutionary Algorithms*) têm sido aplicados para problemas de otimização multi-objetivo (Seção 3.1). O primeiro MOEA implementado foi proposto por SCHAFFER (1985b) e foi denominado VEGA (*Vector Evaluated Genetic Algorithm*). Nesta proposta, Schaffer sugeriu uma modificação no AG para avaliar cada objetivo separadamente. Um dos problemas do algoritmo proposto por Schaffer é que este não obtém boa diversidade nas soluções da *fronteira de Pareto* (Seção 3.1.1).

GOLDBERG (1989) cita um procedimento que ordena as soluções baseado no conceito de dominância e que fornece um valor de aptidão para uma solução proporcional ao número de soluções que esta domina. Com isto, as soluções não dominadas possuem maior aptidão e assim terão maior quantidade de cópias na lista de soluções. Com o objetivo de manter a diversidade das soluções, Goldberg sugeriu a utilização de um método de compartilhamento que calcula o nicho de cada solução dentro da fronteira que a solução pertence. Com base nas ideias iniciais de Goldberg, foram então propostos vários modelos de MOEAs.

O operador de seleção é a principal diferença entre os AEs tradicionais e os MOEAs, quando a comparação entre duas soluções deve se realizar de acordo com o conceito de dominância de Pareto. A Tabela 3.1 sintetiza os principais modelos de MOEAs encontrados na literatura. Em algumas propostas, como MOGA e SPEA, o valor de aptidão é proporcional à dominância da solução. Em outros métodos, como NPGA, utilizam a dominância Pareto e estes não calculam um valor de aptidão.

Os modelos de MOEA são classificados por (DEB, 2001) em dois tipos:

1. Não elitistas: compreende os modelos que, como o próprio nome indica, não utilizam alguma forma de elitismo nas suas interações.

2. Elitistas: compreende os modelos que empregam alguma forma de elitismo. Estudo realizado por ZITZLER; DEB; THIELE (2000) conclui que o elitismo melhora as soluções encontradas por um modelo de MOEA.

Dentre os MOEAs, detalhar-se-á na Seção 3.4.1 o modelo proposto para o NSGA-II. Já na Seção 3.4.2 é descrito o algoritmo SPEA2. O detalhamento de ambos MOEAs justifica-se pela comparação da aplicação dos mesmos no PSP pelo 3PG.

3.4.1 Algoritmo NSGA-II

O algoritmo NSGA-II é baseado em uma ordenação elitista por não-dominância (DEB et al., 2000). O NSGA-II, com a população de indivíduos pais P, gera a população de indivíduos filhos Q como nos AEs convencionais. Na primeira iteração, gera-se uma população P_t, que é ordenada por não-dominância (Seção 3.1.1). Depois, aplicando os

Tabela 3.1: Alguns exemplos de modelos de MOEA.

Sigla	Nome do Modelo	Autores
VEGA	Vector Evaluated Genetic Algorithm	(SCHAFFER, 1985b)
WBGA	Weight Based Genetic Algorithm	(HAJELA; LIN, 1992)
MOGA	Multiple Objective Genetic Algorithm	(FONSECA; FLEMING, 1993)
NSGA	Non-Dominated Sorting Genetic Algorithm	(SRINIVAS; DEB, 1994)
NPGA	Niched-Pareto Genetic Algorithm	(HORN; NAFPLIOTIS; GOLDBERG, 1994)
PPES	Predator-Prey Evolution Strategy	(LAUMANNS; G.; H., 1998)
REMOEA	Rudoph's Elitist Multi-Objective Evolutionay Algorithm	(RUDOLPH, 2001)
NSGA-II	Elitist Non-Dominated Sorting Genetic Algorithm	(DEB et al., 2000)
SPEA, SPEA-2	Strenght Pareto Evolutionary Algorithm 1 e 2	(ZITZLER; THIELE, 1998), (ZITZLER; LAUMANNS; THIELE, 2001)
TGA	Thermodynamical Genetic Algorithm	(KITA et al., 1996)
PAES	Pareto-Archived Evolutionary Strategy	(KNOWLES; CORNE, 1999)
MONGA-I, MONGA-II	Multi-Objective Messy Genetic Algorithm	(VELDHUIZEN, 1999)
Micro-GA	Multi-Objective Micro-Genetic Algorithm	(COELLO; PULIDO, 2001)
PESA-I, PESA-II	Pareto Envelope-Base Selection Algorithm	(CORNE; KNOWLES; OATES, 2000), (CORNE et al., 2001)
RDGA	Rank-Density-based Genetic Algorithm	(HAIMING; GARY, 2003)
GENMOP	General Multi-objective Parallel Genetic Algorithm	(KLEEMAN; LAMONT, 2005)
SDMOGA	Multi-Objective Genetic Algorithm based on Objective Space Divided	(WANGSHU; CHEN; CHEN, 2006)
RJGGA	Real-coding Jumping Gene Genetic Algorithm	(RIPON; SAM; MAN, 2007)

operadores de seleção por torneio (Seção 3.3.4), cruzamento e mutação, obtém-se a população de indivíduos filhos Q_t. Tanto P como Q são de tamanho N.

Para o próximo passo, ambas as populações são unidas em uma nova população $R_t = P_t \bigcup Q_t$, com $|R| = 2N$. Para as seguintes gerações, $n = 1, 2, \ldots,$ o algoritmo NSGA-II trabalha com a população R_t (Figura 3.2).

Obtida a população R_t, realiza-se então a ordenação por não-dominância da mesma, obtendo-se as fronteiras F_1, F_2, ... e todos estes conjuntos são inseridos na nova população P_{t+1}. Considerando que apenas N soluções podem ser inseridas na população P_{t+1}, N soluções de R_t são descartadas. Para preencher as P_{t+1}, começa-se com as soluções em F_1; se não forem completadas as N soluções, prossegue-se com F_2 e, assim por diante. Cada conjunto F_i deve ser inserido na sua totalidade em P_{t+1}, isto ocorre quando $|P_{t+1}| + |F_i| \leq N$. Quando ocorre o caso de ao inserir F_j a $|F_j| > N - |P_{t+1}|$, o algoritmo NSGA-II seleciona então as soluções de F_j que estejam melhor diversificadas. A Figura 3.2 ilustra uma iteração do algoritmo NSGA-II.

O algoritmo NSGA-II emprega um método chamado de distância de multidão (Ver Seção 3.4.1) (*crowding distance*). Tendo obtidas as distâncias, os conjuntos de soluções F_j são ordenados decrescentemente em relação às suas distâncias, e copia-se as primeiras $N - |P_{t+1}|$ soluções de F_j para P_{t+1}. Finalmente, obtém-se Q_{t+1} a partir de P_{t+1} usando os operadores de seleção por torneio, *crossover* e mutação.

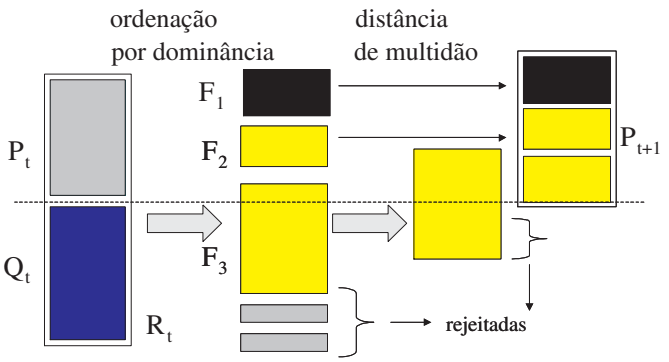


Figura 3.2: Esquema do Modelo NSGA-II (DEB, 2001).

Distância de Multidão

A distância de multidão d_i de uma solução i representa uma estimativa do perímetro formado pelo cubóide, cujos vértices são os seus vizinhos. A Figura 3.3 apresenta a distância de multidão para a solução i, onde I_i^m representa a i-ésima solução na lista ordenada pelo objetivo m. I_1^m e I_l^m são os elementos da lista com o menor e o maior valor para um objetivo m. $f_m^{I_{i+1}^m}$ e $f_m^{I_{i-1}^m}$ são os valores dos vizinhos de i na m-ésima função objetivo. Os f_m^{max} e f_m^{min} são parâmetros dos limites máximo e mínimo em cada objetivo. Quanto maior o cubóide de i, mais afastada se encontra a solução i dos seus vizinhos. As soluções extremas em cada objetivo, ou seja, a melhor e a pior solução em cada objetivo, terão um cubóide infinito.

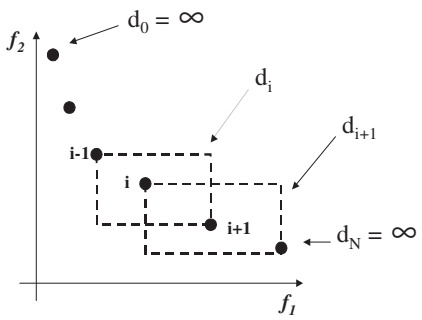


Figura 3.3: Cálculo da distância de multidão no NSGA-II (DEB, 2001).

A forma como é mantida a diversidade entre as soluções não dominadas é a principal vantagem do NSGA-II. O método de comparação por multidão é utilizado para a seleção por torneio e para escolher os elementos da fronteira F_j (DEB, 2001). Se o conjunto F_1 tem um tamanho maior que N, será então executado o processo de escolher apenas N soluções, pois utilizando-se a distância de multidão faz com que sejam perdidas algumas soluções. Seja um F_1 onde existam várias soluções Pareto-ótimas muito próximas e alguma solução distante não Pareto-ótima, mas não dominada no momento. Considerando que o cubóide da solução não dominada é maior, esta solução será copiada em P_{t+1}, enquanto que uma solução Pareto-ótima é eliminada. Esta situação faz com que o NSGA-II possa cair em

um ciclo de gerar soluções Pareto-ótimas e não Pareto-ótimas até convergir finalmente a um conjunto de soluções Pareto-ótimas (DEB, 2001).

3.4.2 Algoritmo SPEA2

O *Strength Pareto Evolutionary Algorithm* 2 (SPEA2), proposto em ZITZLER; LAUMANNS; THIELE (2001), é uma modificação do algoritmo SPEA (ZITZLER; THIELE, 1998). Assim como o NSGA-II, este emprega o conceito de elitismo. Porém, ele usa uma população externa chamada *Archive*, a qual tem a finalidade de armazenar todas as soluções não dominadas até a geração atual.

Para iniciar o algoritmo, é necessário informar o tamanho da população P, ou seja o N_p, além do tamanho da população *Archive* A, o N_a, juntamente com o número de gerações, denotado por T. A população inicial, P_0, deve ser criada, por exemplo aleatoriamente. Já a população *Archive* fica vazia.

Em cada iteração, o *fitness* é calculado para cada uma das soluções $i \in P_t \bigcup A_t$. As soluções não dominadas $i \in P_t \bigcup A_t$ são copiadas para a população *Archive* A_{t+1}. Nesta operação, podem ocorrer três situações, a saber:

1. O valor de N_a é igual ao número de soluções não dominadas: Todas as soluções não dominadas são copiadas para a população *Archive* A_{t+1}.

2. O valor de N_a é menor que o número de soluções não dominadas: Todas as soluções não dominadas $P_t \bigcup A_t$ são inseridas na população *Archive* A_{t+1}. A parte restante é preenchida por meio das soluções dominadas as quais são ordenadas decrescentemente pelo seu valor de *fitness* e, então, as primeiras soluções são copiadas para a população *Archive* A_{t+1} até o preenchimento em sua totalidade.

3. O valor de N_a é maior que o número de soluções não dominadas: Nesta etapa é necessário realizar uma operação de truncamento. Esta operação é descrita em detalhes na sequência desta subseção.

Por fim, é aplicado o processo de seleção por torneio e os operadores genéticos utilizando as populações P_t e A_{t+1} e, assim, a nova população P_{t+1} é criada, reiniciando-se então o processo para criação da população A_{t+2}.

As duas características do SPEA2 são a sua função de *fitness* e a operação de truncamento, as quais serão discorridas nos próximos itens.

Função de *Fitness*

Para a obtenção do valor de *fitness*, o SPEA2 utiliza dois critérios: conceitos de dominância e de densidade. A Equação (3.1) representa a sua função de *fitness*.

$$F(i) = R(i) + D(i) \tag{3.1}$$

onde o primeiro termo é obtido oriundo a Equação (3.2) e o segundo termo por meio da Equação (3.4).

O termo $R(i)$ tem a finalidade de determinar o número de indivíduos que dominam o indivíduo i. Por sua vez, o objetivo é minimizar esse termo, sendo assim, há a relação de que quanto menor o seu valor, melhor o indivíduo i se encontra adaptado. Ou seja, quando $R(i) = 0$ indica que o indivíduo i não é dominado por ninguém, sendo este um indivíduo não dominado. Desta forma, quando seu valor for maior que zero ($R(i) > 0$) significa que i é um indivíduo dominado. A Equação (3.2) é empregada para obter $R(i)$.

$$R(i) = \sum_{j \in P_t + A_t, j \succ i} S(j) \tag{3.2}$$

onde $S(j)$ indica a robustez de cada indivíduo das populações P_t e A_t, ou seja:

$$S(i) = |j | j \in P_t + A_t \wedge i \succ j| \tag{3.3}$$

onde "$|.|$" ilustra a cardinalidade do conjunto, "+" define a união de conjuntos e, por fim, o símbolo "$\succ$" é a relação de dominância de Pareto.

Já o segundo termo da Equação (3.1), ou seja $D(i)$, é empregado para estimar a densidade. O critério da densidade foi empregado para ser um classificador entre as soluções não dominadas quando estas forem em grande quantidade. O valor k é uma adaptação do algoritmo de SILVERMAN (1986), no qual o *k-ésimo* vizinho mais próximo é obtido de maneira mais simplificada, i. e., $k = \sqrt{N_p + N_a}$. Para cada indivíduo i, as distâncias, no espaço dos objetivos, entre i e todos os indivíduos j das populações P_t e A_t são calculadas e armazenadas em uma lista a qual é ordenada crescentemente, e o *k-ésimo* elemento representa o termo σ_i^k. Então o termo $D(i)$ é obtido por:

$$D(i) = \frac{1}{\sigma_i^k + 2} \tag{3.4}$$

Algoritmo de Truncamento

O Algoritmo de Truncamento tem o objetivo de restringir o tamanho de N_a em $t+1$ para N_a. Desta forma, para cada iteração remove-se a solução em que sua distância para o vizinho mais próximo seja a menor entre as distâncias existentes. Caso ocorra empate, é então empregado a segunda distância e assim por diante. Assim, o indivíduo i será removido se $i \leq j$, para todo $j \in P_{t+1}$.

3.5 Considerações Parciais

Neste capítulo foi explando, teoricamente, as abordagens computacionais que serão investigadas, além de tratar também sobre o problema de otimização multi-objetivo como um todo.

Foi possível evidenciar os aspectos do método evolutivo multi-objetivo, enfatizando-se o NSGA-II e o SPEA2. Tais algoritmos constituem a metodologia proposta neste trabalho.

Capítulo 4

Metodologia Proposta

A metodologia proposta fundamenta-se no desenvolvimento do ProtPred-PEO-GROMACS, ou 3PG, um *framework* de computação evolutiva multi-objectivo para a predição *ab initio* da estrutura terciária de proteínas. A abordagem evolutiva multi-objetivo enfatiza-se pelo uso do *framework* ParadisEO e a obtenção das propriedades físicas da proteína é realizada pelo *framework* GROMACS. Já o ProtPred tem a responsabilidade de integrar ambos *frameworks*, tanto para a execução dos algoritmos para a predição quanto para a análise de suas performances.

Para se evidenciar um melhor entendimento da metodologia proposta neste trabalho, as subseções a seguir visam discorrer com maior ênfase os principais tópicos, sendo eles os seguintes: a Subseção 4.1 descreve o *software* ProtPred, o precursor do *framework* aqui proposto. As principais modificações realizadas no ProtPred a fim de torná-lo ProtPred-PEO-GROMACS encontram-se na Subseção 4.2.

4.1 Apresentando o ProtPred

O ProtPred (LIMA et al., 2007); (BRASIL; DELBEM; BONETTI, 2011) é um *software* para investigar o PSP no qual se permite modelá-lo como um problema de otimização. Inclusive, existe a versão do mesmo tratando o PSP em um modelo HP (GABRIEL; MELO; DELBEM, 2012).

Em termos de problema de otimização, ele emprega os AEs sob a ótica mono e multi-objetivo. A Equação (4.1) representa a função de avaliação empregado no AE mono-objetivo.

$$Objetivo = W_1 * E_{bond} + W_2 * E_{angular} + W_3 * E_{dihe} + W_4 * E_{impr} + W_5 * E_{vdw} + W_6 * E_{elec} \quad (4.1)$$

onde os valores $W_{1..6}$ representam os pesos que necessitam ser informados. Os quatro primeiros termos representam as ligações covalentes (estiramentos das ligações covalentes,

ângulos de torsão, Urey-Bradley e Imprópria) e os dois últimos as ligações não-covalentes (van der Waals e eletrostática).

Por outro lado, quando o ProtPred emprega a ótica multi-objetivo, é então utilizado o algoritmo NSGA-II para predizer a estrutura terciária da proteína. Neste caso, a função de avaliação é composta pelos objetivos descritos na Seção 2.8. Em virtude do NSGA-II ser um MOEA, para garantir uma boa acurácia do mesmo, a literatura vem então sugerindo que o empregue com no máximo três objetivos. Assim sendo, os objetivos foram divididos como se segue:

$$Objetivo_1 = E_{angular} + E_{bond} + E_{dihe} + E_{impr} \tag{4.2}$$

$$Objetivo_2 = E_{vdw} \tag{4.3}$$

$$Objetivo_3 = E_{elec} \tag{4.4}$$

onde a Equação (4.2) refere-se ao agrupamento das ligações covalentes (estiramentos das ligações covalentes, ângulos de torsão, Urey-Bradley e Imprópria). A Equação (4.3) é referente à energia de van der Waals e, por fim, a Equação (4.4) refere-se à energia Eletrostática.

Normalmente, para se construir um AE, devem-se seguir cinco passos fundamentais:

1. Como representar os indivíduos;
2. Decidir como será inicializada a população;
3. Formular uma maneira de avaliar os indivíduos (Função de Avaliação ou *Fitness*);
4. Desenvolver os operadores de mutação e recombinação;
5. Decidir qual a estratégia para selecionar os indivíduos.

O primeiro passo analisa como os indivíduos serão representados. Tal passo é dependente do problema em que o AE está sendo empregado. Logo, os indivíduos podem ser representados por matrizes, grafos, valores discretos e outros. Para o PSP, a representação dos indivíduos devem ser os ângulos ϕ e ψ do *backbone*, assim como os ângulos da cadeia lateral (rotâmeros), os quais representam os parâmetros livres das proteínas.

ProtPred utiliza um modelo *full-atom* com coordenadas internas para representar as proteínas. Este foi baseado no fato de que cada aminoácido requer um número fixo de ângulos torsionais para determinar as coordenadas da estrutura terciária de todos os átomos. O comprimento da ligação e os ângulos são considerados em seus valores ideais. Então, para representar uma solução (indivíduo) é necessário os ângulos do *backbone* (ϕ

e ψ) e os valores dos ângulos da cadeia lateral (χ_i) $i = 0, \ldots, 4$, dependendo de cada tipo de resíduo (CUI; CHEN; WONG, 1998).

Já o passo seguinte, discute a metodologia de como inicializar a população, que pode ser aleatoriamente ou baseada em informações. Esta última é conhecida como heurística. Em sua maioria, para o PSP, a inicialização da população é realizada de maneira aleatória.

Tendo o objetivo de reduzir o espaço conformacional, os valores dos ângulos torsionais do *backbone* são restritos em regiões baseadas no CADB-2.0 (MOHAN et al., 2005), o qual contem os valores mais comuns de ângulos torsionais para cada resíduo. Já os ângulos torsionais da cadeia lateral são restritos em regiões derivadas da biblioteca Tuffery (TUFFERY et al., 1991).

A avaliação dos indivíduos (passo três), conhecido como *Fitness* ou função de avaliação, para aplicações reais é a etapa mais custosa computacionalmente. Uma subrotina ou mesmo um processo externo, podem ser empregados para avaliar um indivíduo. Esta etapa, além de dependente do problema em que o AE está sendo utilizado, evidencia as informações que estão sendo empregadas para se buscar os melhores indivíduos. Ou seja, normalmente, utiliza-se esta etapa para distinguir a acurácia dos AEs empregados no mesmo problema. O ProtPred vem utilizando as rotinas do TINKER (PONDER, 2001) e seus parâmetros são oriundos ao campo de força CHARMM (versão 27).

O passo quatro trata a respeito dos operadores genéticos: mutação e recombinação. É neste passo onde se encontra a "inteligência" do algoritmo, uma vez que com os operadores obtêm-se os novos indivíduos da população. Com o operador recombinação, os filhos herdam de algumas partes de seus pais. Esta operação depende de como foi elaborado a representação dos indivíduos (Passo um). Por outro lado, o operador de mutação permite avaliar todo o espaço de busca. Porém, seu valor é importante e deve ser bem controlado. Tais operadores produzem indivíduos válidos, ou seja, com eles é criado novos indivíduos, os quais necessitam ser avaliados. A diferença é, com operador recombinação, os indivíduos criados são parecidos (próximos) de seus pais. Já com o operador mutação, os novos indivíduos podem sofrer alterações aleatórias. Portanto, após esta etapa, faz-se imprescindível avaliar as alterações ocorridas. Vale ressaltar que, para o PSP, os novos indivíduos são as novas conformações da proteína.

Finalmente, o último passo é elaborar a estratégia de seleção dos indivíduos. O AE trabalha com o princípio da Teoria da Evolução, i. e., os indivíduos mais adaptados sobrevivem, sendo que o restante é descartado. Assim sendo, a pressão de seleção guia a evolução da população. Todavia, para garantir uma boa acurácia do algoritmo, os piores indivíduos não devem ser descartados por completo, mas sim, possuir uma chance menor de serem selecionados.

Mais especificamente, tem-se os seguintes passos para a execução do ProtPred: Inicia-se o *loop* principal do algoritmo. Baseando-se nos indivíduos atuais, são obtidos então os novos indivíduos pela aplicação dos operadores genéticos. Neste sentido, o ProtPred

trabalha com três operadores de recombinação. O primeiro é o BLX-α, o qual foi desenvolvido para trabalhar especificamente com os indivíduos cujas representações são realizados por números reais (ponto flutuante) (DEB, 2001). O segundo usa o crossover uniforme sendo que o último operador, emprega dois pontos. Para a mutação também há três operadores. O primeiro atua sobre a cadeia de peptídeos, sendo então alterado todo o *backbone* e rotâmeros (conformação da proteína) aleatoriamente; porém, baseada em suas correspondentes regiões de restrições. Os dois outros operadores aplicam a mutação uniforme. Eles modificam todos os valores do *backbone* e rotâmeros dos resíduos selecionados. A diferença entre o segundo e o terceiro operador é a distribuição uniforme. Para o segundo, a distribuição varia entre 0 e 1. Para o terceiro, o intervalo é 0 e 0.1.

4.2 Modificando o ProtPred

O ProtPred vem sendo aplicado no PSP. Porém, o mesmo possui algumas limitações:

1. A sua execução é dependente, exclusivamente, do campo de força CHARMM em sua versão 27. Ou seja, todos os parâmetros são oriundos à esse campo de força e necessita ser informado manualmente, já que não possui integração com o mesmo;

2. A computação das propriedades físicas da proteínas (função de avaliação) é de implementação própria. Ou seja, não é integrado a nenhum *framework* de modelagem molecular. Desta forma, é necessário implementar todos os objetivos;

3. A implementação computacional dos conceitos da computação evolutiva é necessária mesmo se esses já tenham sido desenvolvidos em algum *framework* específico de computação evolutiva.

Em virtude da complexidade e heterogeneidade dos problemas da modelagem molecular para proteínas, esta exige que seus algoritmos trabalhem com mais de um campo de força, pois cada um deles possuem propriedades específicas. Além disso, é enfatizado para tais algoritmos a empregarem técnicas para reduzir seu custo computacional, para assim, conseguirem realizar suas tarefas eficientemente. Portanto, não há uma única técnica computacional a fim de explorar eficientemente o espaço conformacional dos problemas da modelagem molecular. Por conseguinte, possuir uma infraestrutura para investigar o emprego de técnicas computacionais torna-se extremamente recomendado.

De fato, o campo da modelagem molecular é cada vez mais útil nas pesquisas básicas tal como a biotecnologia. Entretanto, a ausência de um *framework user-friendly* o qual providencia um acesso à uma gama de informações moleculares, vem desencorajando a adoção de métodos computacionais por não especialistas em computação, os quais possuem muita experiência na área experimental (SAREL et al., 2011). Todos os *frameworks* disponíveis proporcionam algumas informações sobre a proteína, mas não todas. Portanto,

é necessário trabalhar com mais de um *framework* tornando uma situação desconfortável uma vez que os mesmos podem trabalhar com diferentes campos de forças e/ou unidades.

Dentre as técnicas computacionais, as quais vêm sendo empregadas nestes tipos de problemas, destacam-se os Algoritmos Evolutivos (AE). Os AEs são fáceis de se empregar nos problemas de otimização multi-objetivo e mono-objetivo (JAIMES; COELLO, 2008). Visto que os AEs são empregados em diversos tipos de problemas, a literatura vem reportando o desenvolvimento de *frameworks*, os quais implementam os conceitos genéricos dos AEs (técnicas de seleção, operadores genéticos, Dominância de Pareto por exemplo), para assim obter uma padronização dos AEs empregados.

Mais especificamente, o PSP sendo um problema complexo e sem uma solução plausível, os pesquisadores vem aplicando diferentes técnicas e métodos. Este cenário é propício à adoção de *frameworks* especializado para o PSP. Neste sentido destacam-se os trabalhos SIMONS et al. (1999), HONIG (1999), KLEPEIS; FLOUDAS (2003) e ZHANG (2008a). Nestes trabalhos não é permitido modificar a técnica computacional. Ou seja, os pesquisadores podem simplesmente executar o *framework* alterando os parametros para o algoritmo proposto no mesmo. Além disso, não se utilizam de computação evolutiva, embora os algoritmos evolutivos multi-objetivo vem sendo difundido seu uso, pois no PSP envolve um compromisso entre os diferentes objetivos de acordo com o conflito cenário do funil (Seção 2.7). Desta forma, uma solução ótima em um dos objetivos pode não ser ótima no outro (CUTELLO; NARZISI; NICOSIA, 2006a; HANDL; LOVELL; KNOWLES, 2008).

É possível aqui compreender a necessidade do ProtPred em ser aperfeiçoado a fim de atender as exigências da modelagem molecular no que tange a ser um promissor preditor de estrutura terciária da proteína, a partir somente de sua sequência primária em um ambiente de simulação integrado, o qual se torna possível investigar modelos físicos com algoritmos populacionais. Para tanto, foi escolhido aqui dois projetos computacionais *open-source* os quais são difundidos na literatura. Estes projetos são o ParadisEO e o GROMACS. Mais especificamente, o ProtPred será integrado à estes projetos. Desta integração surgiu o *framework* ProtPred-PEO-GROMACS, o qual será aplicado no PSP. As principais modificações do ProtPred-PEO-GROMACS (3PG) em relação ao ProtPred, as quais também são partes das contribuições desta tese, são as seguintes:

- O ProtPred-PEO-GROMACS é integrado com o GROMACS (SPOEL et al., 2009) e não é implementações baseadas no Tinker. Esta característica é importante, pois GROMACS é muito rápido e eficiente para a computação das propriedades da proteína, por exemplo: energia potencial, área de acessibilidade ao solvente, o número de ligações de hidrogênio e o raio de giro. Além disso, com a integração, torna-se o emprego das propriedades da proteína de forma mais fácil já que as mesmas encontram-se implementadas no GROMACS;

- O algoritmo de conversão de coordenadas internas para cartesiano é realizado pela

implementação do SN-NeRF (PARSONS et al., 2005);

- O *framework* constrói a topologia atomística baseando-se na sequência primária contendo as informações: átomos e seus aminoácidos, como também, suas cargas e os quatro átomos que formam cada ângulo diedral do *backbone* e da cadeia lateral;

- Da mesma maneira, o 3PG estando integrado com o *framework* ParadisEO, o desenvolvimento de algoritmos evolutivos, além de padronizados, torna-se mais rápido, uma vez que o mesmo já se encontra implementado no ParadisEO.

A representação das soluções (indivíduos) no ProtPred-PEO-GROMACS é a mesma já empregada no ProtPred. Porém, o algoritmo de conversão de coordenadas internas para Cartesianas é realizado pela implementação do SN-NeRF (PARSONS et al., 2005). Visto que as proteínas são polipeptídeos é então necessário empregar a Matriz Z que armazena as informacões acerca da topologia da proteína em detalhes atomístico como por exemplo: as distâncias entre os átomos ligados, os valores dos ângulos planares e os valores dos ângulos diedrais. A representação Cartesiana é um *full-atom*, incluindo todos os átomos de Hidrogênios baseados na implementação do campo de força CHARMM27 no GROMACS (BJELKMAR et al., 2010).

Na aquisição das propriedades físicas, energéticas e estruturais da proteína é empregado o GROMACS. Ele possui um conjunto de *softwares* para a modelagem molecular, como, por exemplo, o g_sas, que calcula a área de acessibilidade hidrofóbica, hidrofílica e total com o solvente de acordo com EISENHABER et al. (1995) e EISENBERG; MCLACHLAN (1986). A energia potencial é obtida por meio do programa g_energy. Já os programas g_hbond e g_gyrate computam, respectivamente, as ligações de hidrogênio e o raio de giro.

A inicialização da população inicial é realizada da mesma forma que o ProtPred. Ou seja, os ângulos diedrais de cada resíduo do indivíduo são oriundos as bibliotecas CADB 2.0 e Tuffery.

Todas essas modifcações ocorreram para integrar o ProtPred à um *framework* de modelagem molecular. Dentre os *frameworks*, foi escolhido o GROMACS. Logo, torna-se possível o desenvolvimento de um *framework* evolutivo mono-objetivo. Sendo assim, esta etapa caracteriza o ProtPred-GROMACS.

Já a integração com o ParadisEO permite ao ProtPred-GROMACS empregar o conceito multi-objetivo. Este conceito é representado por dois algoritmos evolutivos multi-objetivos, a saber: NSGA-II e SPEA2, os quais encontram-se respectivamente detalhados as Subseções 3.4.1 e 3.4.2. Confeccionou-se então o ProtPred-PEO-GROMACS ou 3PG. A Figura 4.1 ilustra o diagrama do *framework* proposto. Assim, é possível notar que a capacidade de alavancar novas implementações de propriedades das proteínas é independente do desenvolvimento de novos algoritmos para serem aplicados no PSP. Desta forma, o ProtPred tem a responsabilidade de unir os *frameworks* GROMACS e ParadisEO.

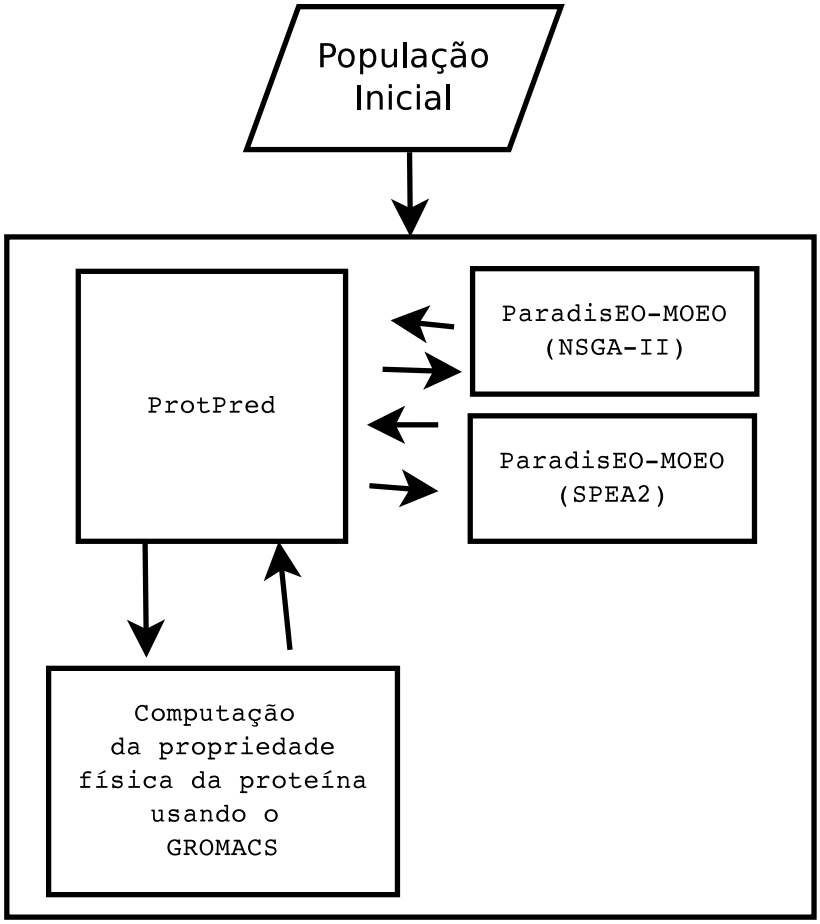


Figura 4.1: Representação do Diagrama do *framework* proposto.

4.3 Considerações Parciais

Neste capítulo foi possível descrever de forma resumida o método proposto e proporcionar o cenário de aplicação para o *framework* 3PG. Além disso, foi apresentado o histórico do ProtPred, sendo este o projeto base de motivação para o desenvolvimento deste trabalho.

Os resultados da aplicação do *framework* proposto no problema de predição de estrutura terciária de proteínas, assim como detalhes de seu desenvolvimento, serão apresentados no Capítulo 5.

Capítulo 5

Resultados e Aspectos de Desenvolvimento do 3PG

O desenvolvimento do *framework* proposto foi realizado em duas etapas, as quais são discorridas com mais ênfase nas Seções 5.1 e 5.4. Na Seção 5.1 é descrito a integração entre o algoritmo ProtPred e o GROMACS, originando o ProtPred-GROMACS. A avaliação da performance da reconstrução da proteína de coordenadas internas para Cartesianas é enaltecido na Seção 5.2. A aplicação do ProtPred-GROMACS no PSP encontra-se descrito na Seção 5.3. Já na Seção 5.4 é enfatizado a integração do ProtPred-GROMACS com o ParadisEO. Desta integração, originou-se então o ProtPred-PEO-GROMACS ou 3PG. Em sequência, é apresentado na Seção 5.5 a aplicação do *framework* proposto no PSP.

5.1 Integração do ProtPred com GROMACS

Nesta seção é descrita a implementação da integração do ProtPred com o GROMACS. Conforme enfatizado na Subseção 4.2 o sistema ProtPred necessitou sofrer alterações a fim de atender às exigências da modelagem molecular. Em uma delas enfatiza-se a substituição da sua implementação de função de avaliação baseadas no TINKER para o GROMACS.

Para a execução de um AE, a primeira etapa é a criação da população inicial. Esta etapa no *framework* proposto é de responsabilidade do *software* protpred-Gromacs_pop_initial. A obtenção da sequência primária da proteína é obtida por meio do arquivo Fasta que é um padrão de arquivo em modelagem molecular. Neste padrão, encontram-se disponíveis as informações a respeito da disposição dos resíduos da proteína, ou seja sua sequência primária. A Figura 5.1 ilustra o arquivo Fasta da proteína 1VII, onde as informações são dispostas baseando-se no cabeçalho do arquivo. Assim 1VII:A enfatiza, respectivamente, o PDBID e o nome da cadeia, que por sua vez, a linha abaixo, indica a sequência primária da proteína.

```
>1VII:A|PDBID|CHAIN|SEQUENCE
MLSDEDFKAVFGMTRSAFANLPLWKQQNLKKEKGLF
```

Figura 5.1: Representação do arquivo Fasta da proteína 1VII.

A topologia atomística da proteína é construída automaticamente baseando-se na sua sequência primária. Assim, a topologia torna-se útil para a obtenção das informações necessárias para a representação da proteína em coordenadas internas e Cartesianas. Visando um melhor entendimento das informações da topologia, a mesma é separada em seções assim descritas:

1. **General information** contêm informações de propósito gerais: número de átomos total, número de resíduos, número de ligações covalentes, número de ângulos, a carga da proteína e o número de tipos de ângulos diedrais;

2. **Atom** informa os átomos de cada resíduo, o seu nome e a sua carga;

3. **Residue and its sequence atoms** informa o número do átomo inicial e final de cada aminoácido. Desta forma, a busca dos átomos dos resíduos torna-se mais rápida e eficiente computacionalmente, pois a busca em seu pior caso tem ordem de complexidade $O(N)$ onde N é o número de átomos do resíduo;

4. **Residues and atoms for phi angle** contêm os índices dos átomos para o cálculo do ângulo phi (ϕ) de cada resíduo. Estes índices são baseados nos valores da seção **Atom**;

5. **Residues and atoms for psi angle** contêm os índices dos átomos para o cálculo do ângulo psi (ψ) de cada resíduo. Estes índices são baseados nos valores da seção **Atom**;

6. **Residues and atoms for side chains angle** informa os índices dos átomos necessários para calcular os ângulos da cadeia lateral (χ_i) $i = 0, \ldots, 4$, dependendo de cada tipo de resíduo. Estes índices são baseados nos valores da seção **Atom**.

A Figura 5.2 apresenta uma representação da Seção **General Information** da proteína 1VII, a qual permite saber a quantidade de átomos, o número de resíduos, quantidade de ligações, o valor da carga da proteína e a quantidade de números de ângulos diedrais.

```
[General information section]
;num_atom  num_res num_bond_angles num_protein_side_chains protein_charge num_dihedral_angle_type
 596  36 280 79 2 73
```

Figura 5.2: Representação da Seção **General Information** da proteína 1VII.

Uma representação da seção **Atom** da topologia da proteína 1VII está na Figura 5.3a. Na mesma encontram-se as informações atomísticas dos dois primeiros resíduos (MET e LEU) da proteína. É importante observar que o índice dos átomos é único e, assim, o átomo é identificado independentemente do resíduo que o mesmo se encontra. Além disso, os valores de carga de cada átomo são disponibilizados.

Já a Figura 5.3b ilustra o átomo inicial e o final pertencentes a cada aminoácido da proteína. Assim, é mostrada a seção **Residue and its sequence atoms** da topologia da proteína 1VII. Esta informação permite uma melhor performance na procura de um átomo, já que conhecendo o seu resíduo é possível restringir o início e o fim da procura.

```
[Atom section]
;num_atom  atom_name  res_num  res_name charge
1      N    1  MET   -0.300000
2      CA   1  MET    0.210000
3      C    1  MET    0.510000
4      O    1  MET   -0.510000
5      H1   1  MET    0.330000
6      H2   1  MET    0.330000
7      H3   1  MET    0.330000
8      HA   1  MET    0.100000
9      CB   1  MET   -0.180000
10     CG   1  MET   -0.140000
11     SD   1  MET   -0.090000
12     CE   1  MET   -0.220000
13     HB1  1  MET    0.090000
14     HB2  1  MET    0.090000
15     HG1  1  MET    0.090000
16     HG2  1  MET    0.090000
17     HE1  1  MET    0.090000
18     HE2  1  MET    0.090000
19     HE3  1  MET    0.090000
20     N    2  LEU   -0.470000
21     CA   2  LEU    0.070000
22     C    2  LEU    0.510000
23     O    2  LEU   -0.510000
24     HN   2  LEU    0.310000
25     HA   2  LEU    0.090000
26     CB   2  LEU   -0.180000
27     CG   2  LEU   -0.090000
28     CD1  2  LEU   -0.270000
29     CD2  2  LEU   -0.270000
30     HB1  2  LEU    0.090000
31     HB2  2  LEU    0.090000
32     HG   2  LEU    0.090000
33     HD11 2  LEU    0.090000
34     HD12 2  LEU    0.090000
35     HD13 2  LEU    0.090000
36     HD21 2  LEU    0.090000
37     HD22 2  LEU    0.090000
38     HD23 2  LEU    0.090000
```

(a) Seção Atom.

```
[Residue and its sequence atoms]
;res_num  atom_first atom_end
1  1   19
2  20  38
3  39  49
4  50  61
5  62  76
6  77  88
7  89  108
8  109 130
9  131 140
10 141 156
11 157 176
12 177 183
13 184 200
14 201 214
15 215 238
16 239 249
17 250 259
18 260 279
19 280 289
20 290 303
21 304 322
22 323 336
23 337 355
24 356 379
25 380 401
26 402 418
27 419 435
28 436 449
29 450 468
30 469 490
31 491 512
32 513 527
33 528 549
34 550 556
35 557 575
36 576 596
```

(b) Seção Residue and its sequence atoms.

Figura 5.3: Representação da topologia do 3PG referente as seções **Atom** e **Residue and its sequence atoms proteína** 1VII.

A identificação dos quatro átomos necessários para o cálculo do ângulo diedral ϕ são armazenados na seção **Residues and atoms for phi angle** da topologia. Desta forma, a Figura 5.4a ilustra esta seção por meio da proteína 1VII. Para exemplificar, observam-se os átomos do segundo aminoácido. O átomo com identificação 3 pertence ao primeiro aminoácido. Ou seja, com essa seção o cálculo do ângulo ϕ torna-se eficiente, pois não é necessário procurar a posição 3D do átomo do resíduo anterior toda vez que for obtê-lo.

De maneira análoga ao ângulo ϕ, o *framework* proposto armazena na seção **Residues and atoms for psi angle** os quatro átomos necessários para o cálculo do ângulo diedral ψ. Assim, na Figura 5.4b é ilustrada esta seção por meio da proteína 1VII.

Por fim, a seção **Residues and atoms for side chains angle** da topologia da proteína é representada na Figura 5.4c. Salienta-se o emprego desta seção em virtude do número de ângulos *chi* da cadeia lateral depender de cada aminoácido. Conforme é possível observar na figura, o primeiro resíduo (MET) possui três *chi*. Já o segundo (LEU) possui apenas dois ângulos *chi*.

```
[Residues and atoms for phi angle]
;res_num  atom_1 atom_2  atom_3 atom_4
1  0  1  2   3
2  3  20  21   22
3  22  39  40   41
4  41  50  51   52
5  52  62  63   64
6  64  77  78   79
7  79  89  90   91
8  91  109  110   111
9  111  131  132   133
10  133  141  142   143
11  143  157  158   159
12  159  177  178   179
13  179  184  185   186
14  186  201  202   203
15  203  215  216   217
16  217  239  240   241
17  241  250  251   252
18  252  260  261   262
19  262  280  281   282
20  282  290  291   292
21  292  304  305   306
22  306  323  324   325
23  325  337  338   339
24  339  356  357   358
25  358  380  381   382
26  382  402  403   404
27  404  419  420   421
28  421  436  437   438
29  438  450  451   452
30  452  469  470   471
31  471  491  492   493
32  493  513  514   515
33  515  528  529   530
34  530  550  551   552
35  552  557  558   559
36  559  576  577   578
```

(a) Seção ϕ.

```
[Residues and atoms for psi angle]
;res_num  atom_1 atom_2  atom_3 atom_4
1  1  2  3   20
2  20  21  22   39
3  39  40  41   50
4  50  51  52   62
5  62  63  64   77
6  77  78  79   89
7  89  90  91   109
8  109  110  111   131
9  131  132  133   141
10  141  142  143   157
11  157  158  159   177
12  177  178  179   184
13  184  185  186   201
14  201  202  203   215
15  215  216  217   239
16  239  240  241   250
17  250  251  252   260
18  260  261  262   280
19  280  281  282   290
20  290  291  292   304
21  304  305  306   323
22  323  324  325   337
23  337  338  339   356
24  356  357  358   380
25  380  381  382   402
26  402  403  404   419
27  419  420  421   436
28  436  437  438   450
29  450  451  452   469
30  469  470  471   491
31  491  492  493   513
32  513  514  515   528
33  528  529  530   550
34  550  551  552   557
35  557  558  559   576
36  576  577  578   0
```

(b) Seção ψ.

```
[Residues and atoms for side chains angle]
;res_num  chi atom_1 atom_2  atom_3 atom_4
1  1 1  2  9   10
1  2 2  9  10   11
1  3 9  10  11   12
2  1 20  21  26   27
2  2 21  26  27   28
3  1 39  40  45   46
4  1 50  51  56   57
4  2 51  56  57   58
5  1 62  63  68   69
5  2 63  68  69   70
5  3 68  69  70   71
6  1 77  78  83   84
6  2 78  83  84   85
7  1 89  90  95   96
7  2 90  95  96   97
8  1 109  110  115   116
8  2 110  115  116   117
8  3 115  116  117   118
8  4 116  117  118   119
9  1 137  133  132   131
10  1 141  142  147   148
11  1 157  158  163   164
11  2 158  163  164   165
```

(c) Seção chi.

Figura 5.4: Representação da topologia do 3PG referente as seções ϕ, ψ e *chi* da proteína 1VII.

Em se tratando de computação evolutiva, a solução sendo representada, neste trabalho, por uma conformação da proteína, necessita ter suas propriedades (interações) computadas, as quais consistirão dos objetivos. Este cálculo é obtido por meio da integração com o GROMACS. Uma vez que o GROMACS emprega as coordenadas Cartesianas, é então

necessário realizar a conversão de coordenada interna para esta representação. Para isto, é utilizado a implementação descrita no trabalho de PARSONS et al. (2005), onde descreve o algoritmo SN-Nerf (*Self-Normalizing Natural Extension Reference Frame*). Segundo os autores, este algoritmo possui uma melhor performance computacional, além de tentar evitar *overlaps*, pois se utiliza da distância entre o segundo e terceiro átomo, e não somente a distância de ligação do átomo conectado. A Seção 5.2 tem os detalhes desta conversão.

Tendo em vista que a proteína é um polímero e que a cada geração uma nova população terá um conjunto de indivíduos para obterem seus *fitness*, houve o desenvolvimento da matriz Z a qual armazena informações atomísticas. A Figura 5.5 ilustra um exemplo desta matriz para a proteína 1VII. Ressalta-se que a criação da matriz Z, além de automática, é dependente da topologia. O valor da linha 19 do campo Index_Top mostra que o átomo Nitrogênio (pertencente ao segundo aminoácido) liga-se covalentemente com o átomo 3, que é o Carbono pertencente ao primeiro aminoácido, em uma uma distância de 1.30Å. Além disso, este Nitrogênio forma um ângulo de 62.612408 graus com o átomo 2, o qual é o C_α do primeiro aminoácido. Por fim, este átomo forma um ângulo diedro ψ com o átomo Nitrogênio do primeiro resíduo. Logo, o valor desse ângulo diedro encontra-se no indivíduo. Portanto, para este caso, o valor seria -38.84 graus.

```
Index_Top Seq Atom_Reference Atom_Name Atom_Connected Bond_Len Atom_Angle Bond_Angle Atom_Dihedral Dihedral_Angle type_diedhral bond_len_2
0 1 1 N
1 2 2 CA 1 1.490000
2 3 3 C 2 1.530000 70.897377 1.490000
3 4 4 O 3 1.210000 2 58.848072 1 dieh_1 1.490000
4 5 5 H1 1 1.000000 2 70.530685 3 dieh_180 1.490000
5 6 6 H2 1 1.000000 2 70.530685 3 trans_120 1.490000
6 7 7 H3 1 1.000000 2 70.530685 3 trans_240 1.490000
7 8 8 HA 2 1.000000 1 70.530685 3 dieh_0 1.490000
8 9 9 CB 2 1.530000 1 69.499359 3 trans_123_ 1.490000
9 10 10 CG 9 1.520000 2 65.901184 1 CHI1 1.530000
10 11 11 SD 10 1.803000 9 67.299202 2 CHI2 1.520000
11 12 12 CE 11 1.791000 10 69.098289 9 CHI3 1.803000
12 13 13 HB1 9 1.000000 10 70.530685 2 dieh_0 1.520000
13 14 14 HB2 9 1.000000 10 70.530685 2 dieh_180 1.520000
14 15 15 HG1 10 1.000000 11 70.530685 9 dieh_0 1.803000
15 16 16 HG2 10 1.000000 11 70.530685 9 dieh_0 1.803000
16 17 17 HE1 12 1.000000 11 70.530685 10 dieh_180 1.791000
17 18 18 HE2 12 1.000000 11 70.530685 10 trans_120 1.791000
18 19 19 HE3 12 1.000000 11 70.530685 10 trans_240 1.791000
19 20 20 N 3 1.300000 2 62.612408 1 PSI_ 1.530000
20 21 21 CA 20 1.490000 3 60.062740 2 dieh_180 1.300000
21 22 22 C 21 1.530000 20 70.897377 3 PHI 1.490000
22 23 23 O 22 1.210000 21 58.848072 20 dieh_1 1.490000
23 24 24 HN 20 1.000000 3 70.450470 21 dieh_0 1.300000
24 25 25 HA 21 1.000000 20 70.530685 22 dieh_0 1.490000
25 26 26 CB 21 1.540000 20 69.499359 22 trans_123_ 1.490000
26 27 27 CG 26 1.530000 21 63.701031 20 CHI1 1.530000
27 28 28 CD1 27 1.521000 26 69.298820 21 CHI2 1.530000
28 29 29 CD2 27 1.521000 26 66.199127 21 dieh_7 1.530000
29 30 30 HB1 26 1.000000 21 70.530685 27 dieh_0 1.530000
30 31 31 HB2 26 1.000000 21 70.530685 27 dieh_180 1.530000
31 32 32 HG 27 1.000000 26 70.530685 28 dieh_0 1.530000
32 33 33 HD11 28 1.000000 27 70.530685 26 dieh_180 1.530000
33 34 34 HD12 28 1.000000 27 70.530685 26 trans_120 1.530000
```

Figura 5.5: Representação da matriz Z para a proteína 1VII

O não armazenamento dos valores dos ângulos diedrais na matriz Z é justificável, uma vez que são esses ângulos aqueles modificados durante a execução do algoritmo evolutivo, ou seja, a aplicação dos operadores genéticos é realizada nos mesmos. Assim, pode-se criar uma única instância desta matriz, já que a topologia da proteína não se altera ao longo da execução do algoritmo de predição.

5.2 Avaliação da Conversão de Coordenadas Internas para Cartesianas

Conforme descrito na Seção 2.6, as proteínas podem ser representadas computacionalmente por meio das coordenadas internas e coordenadas Cartesianas. A representação das proteínas no 3PG é por meio de coordenadas internas, uma vez que com essa representação torna-se possível empregar os operadores genéticos de números reais já conhecidos na literatura. Entretanto, o GROMACS usa a representação Cartesiana para a proteína. Logo, fica evidente a necessidade de um algoritmo para converter coordenadas internas em coordenadas Cartesianas. Para isso foi utilizado a implementação do algoritmo SN-Nerf (PARSONS et al., 2005).

No artigo de PARSONS et al. (2005) o algoritmo SN-Nerf é comparado com outros três algoritmos de conversão de coordenadas internas para Cartesianas, a saber: General–Rotation (SUH; RADCLIFFE, 1978), Rodrigues–Gibbs (BOTTEMA; ROTH, 1979) e Quaternion (HAMILTON, 1853). Como resultado descreve-se uma melhor performance do SN-Nerf. Por esta razão, o mesmo foi implementado neste projeto. Todavia, é importante ressaltar que o 3PG foi concebido para aplicar outros conversores, os quais podem ser adicionados com poucas alterações em seu código-fonte.

Na Figura 5.6 é apresentado uma iteração do algoritmo SN-Nerf para posicionamento do átomo D. Para isso, é necessário conhecer as coordenadas Cartesianas dos átomos A, B e C. Inicialmente, o átomo D é colocado em D_0, uma distância de ligação CD. Neste ponto ocorre a primeira rotação sobre o átomo C no plano ABC. Como resultado, tem-se a posição D_1 sendo, então, rotacionada sobre eixo de ligação BC e ângulo torsional no qual, finalmente, obtém o ponto D_2. Estes passos são necessários em todos os algoritmos de conversão. Para alavancar um melhor desempenho, o método proposto em PARSONS et al. (2005) posiciona diretamente o átomo D em D_2. Para tanto, o sistema de coordenadas é transformado em um plano XY, o qual está no plano ABC. Logo, esta transformação é aplicada para D_2.

Foram escolhidas quatro proteínas para demonstrar sua performance no que tange a variabilidade de estruturas secundárias. Tais proteínas encontram-se depositadas no *Protein Data Bank* (PDB) e sua identificação: 1VII (MCKNIGHT; MATSUDAIRA; KIM, 1997), 1A11 (OPELLA et al., 1999), 1PLW (MARCOTTE et al., 2004) e 1UAO (HONDA et al., 2004). Em todos os casos, os diedros foram computados utilizando a estrutura nativa de cada proteína. Já os demais parâmetros (ligação, ângulo, etc) são considerados os ideais.

A proteína 1PLW, mesmo com apenas cinco resíduos, é difundida na literatura de predição de sua estrutura terciária por meio de algoritmos evolutivos. Logo, constitui do primeiro teste para mensurar a capacidade da implementação do algoritmo SN-Nerf na reconstrução da estrutura nativa. Por sua vez, a Figura 5.7a ilustra a estrutura nativa

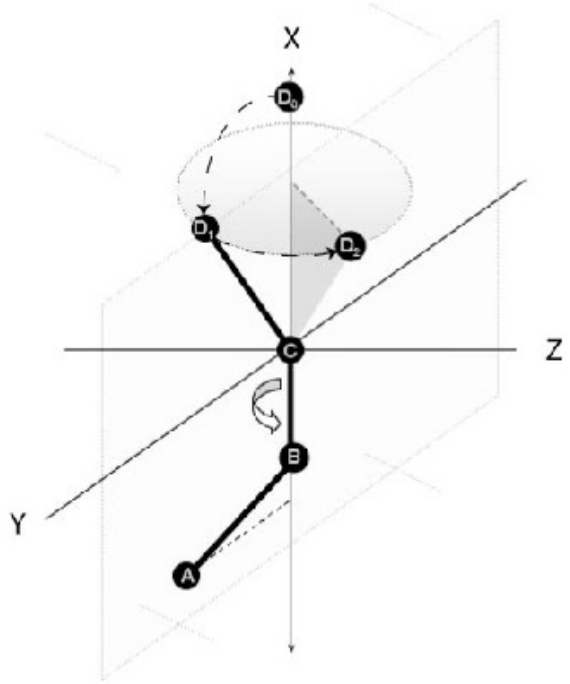


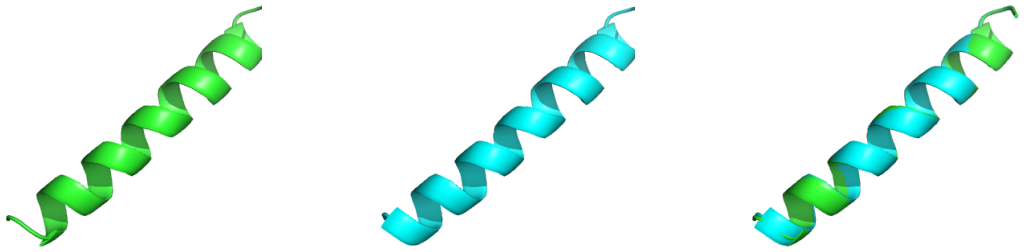
Figura 5.6: Representação do posicionamento de átomo pelo SN-Nerf. Figura baseada em (PARSONS et al., 2005).

da proteína 1PLW. A sua reconstrução é possível visualizar por meio da Figura 5.7b. Finalmente, o alinhamento entre as estruturas com um RMSD de 0.13 Å é apresentado na Figura 5.7c.

(a) Nativa. (b) Reconstruída. (c) Alinhadas.

Figura 5.7: Representação da conversão da estrutura nativa da proteína 1PLW aplicando o algoritmo SN-Nerf e a topologia do 3PG.

Na performance de conversão da proteína 1A11, na qual a estrutura nativa é destacada na Figura 5.8a, é possível notar que há somente hélices como estrutura secundária. Já a Figura 5.8b indica a reconstrução da estrutura nativa por meio do algoritmo SN-Nerf. O alinhamento entre a estrutura e a reconstruída, tendo um valor de RMSD de 0.21Å, é representado na Figura 5.8c.



(a) Nativa. (b) Reconstruída. (c) Alinhadas.

Figura 5.8: Representação da conversão da estrutura nativa da proteína 1A11 aplicando o algoritmo SN-Nerf e a topologia do 3PG.

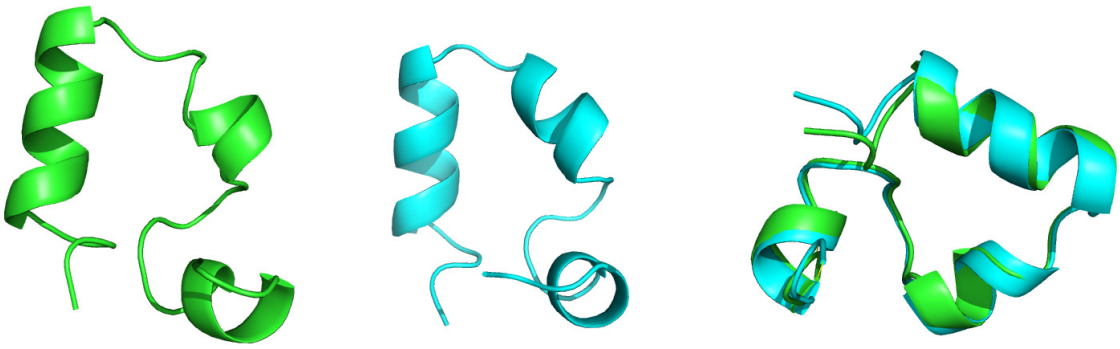
A proteína 1UAO contem dez resíduos possuindo a topologia de folhas Beta. A representação da sua estrutura nativa encontra-se na Figura 5.9a. Por sua vez, a reconstrução desta estrutura por meio do algoritmo SN-Nerf é indicada na Figura 5.9b. Finalmente, alinhando estas estruturas consegue-se um RMSD de 0.14 Å, sendo então destacado na Figura 5.9c.

(a) Nativa. (b) Reconstruída. (c) Alinhadas.

Figura 5.9: Representação da conversão da estrutura nativa da proteína 1UAO aplicando o algoritmo SN-Nerf e a topologia do 3PG.

A proteína 1VII contem trinta e seis resíduos tendo somente hélices como estrutura secundária. As Figuras 5.10a e 5.10b ilustram, respectivamente, a estrutura nativa e a estrutura reconstruída. O alinhamento entre essas estruturas encontra-se na Figura 5.10c, sendo que o RMSD é 0.90Å.

Visto que a aplicação do algoritmo SN-Nerf para reconstruir a representação Cartesiana da estrutura nativa a partir da sua coordenada interna realiza-se de forma satisfatória, na seção a seguir (5.3) discorrerá o emprego do ProtPred-GROMACS na predição da estrutura terciária de proteínas.



(a) Nativa. (b) Reconstruída. (c) Alinhadas.

Figura 5.10: Representação da conversão da estrutura nativa da proteína 1VII aplicando o algoritmo SN-Nerf e a topologia do 3PG.

5.3 Aplicação do ProtPred-GROMACS no PSP

A finalidade desta seção é apresentar a aplicação do ProtPred-GROMACS no problema de predição da estrutura terciária de proteínas. Neste quesito tem-se o desenvolvimento de um algoritmo evolutivo mono-objetivo, a fim de avaliar a eficiência da integração entre o ProtPred e o GROMACS.

Tendo o objetivo de tornar o uso do ProtPred-GROMACS *user-friendly* e, assim, permitir o seu uso por não especialistas em computação, utiliza-se um arquivo de configuração o qual contem todos os parâmetros utilizados pelo algoritmo. Assim, o tamanho da população, o número de gerações e as opções de objetivos tornam-se facilmente acessíveis.

ProtPred-GROMACS para avaliar a estrutura da molécula, nesta aplicação, significa a energia potencial e a área hidrofóbica acessível ao solvente, necessitando ser representada em Coordenadas cartesianas, uma vez que tal representação é empregada no GROMACS que não se utiliza de coordenadas internas.

Neste teste foi avaliada a execução de um algoritmo mono-objetivo implementado no *framework* protpred-GROMACS. Três proteínas foram escolhidas e sua performance foi confrontada. Tais proteínas encontram-se depositadas no *Protein Data Bank* (PDB) e suas identificações são as seguintes: 1VII (MCKNIGHT; MATSUDAIRA; KIM, 1997), 1PLW (MARCOTTE et al., 2004) e 1UAO (HONDA et al., 2004).

Na Tabela 5.1 é mostrado o RMSD final de cada proteína utilizando como *fitness* a energia Potencial. O número de gerações foi 1000.

Na Tabela 5.2 é mostrado o RMSD final de cada proteína utilizando como *fitness* a área hidrofóbica acessível ao solvente. O número de gerações foi 1000.

Por fim, a Tabela 5.3 ilustra os melhores RMSDs obtidos e os compara com outras abordagens da literatura.

Tabela 5.1: RMSDs finais (Å) calculados utilizando a energia Potencial como *fitness*. O número de gerações foi 1000.

PDBID	Indivíduos	RMSD
1VII	100	6.68841
1VII	200	5.4987
1PLW	100	0.66751
1PLW	200	0.41534
1UAO	100	2.98747
1UAO	200	2.92967

Tabela 5.2: RMSDs finais (Å) calculados utilizando a área hidrofóbica acessível ao solvente como *fitness*. O número de gerações foi 1000.

PDBID	Indivíduos	RMSD
1VII	100	7.79486
1VII	200	6.50174
1PLW	100	0.82993
1PLW	200	0.78115
1UAO	100	3.49334
1UAO	200	2.85248

Tabela 5.3: Melhores RMSDs Å obtidos pelo ProtPred-GROMACS e o seu valor correspondente de RMSD nas predições encontradas na literatura.

PDBID	*Framework* proposto	Literatura
1VII	5.5	4.7 - 7.4
1PLW	0.4	1.891
1UAO	2.9	0.34 - 2.50 ; 2.0 - 3.5

5.4 Integração do ProtPred-GROMACS com ParadisEO

O ParadisEO é um *framework open-source* de meta-heurísticas. Ele provê vários componentes reutilizáveis, os quais aceleram e minimizam os esforços na implementação de algoritmos meta-heurísticos. O ParadisEO é composto por quatro principais módulos:

1. **ParadisEO-EO**: para o desenvolvimento de algoritmos meta-heurísticos populacionais.

2. **ParadisEO-MO**: focado no desenvolvimento de algoritmos meta-heurísticos com um único objetivo (mono-objetivos).

3. **ParadisEO-MOEO**: contem os otimizadores meta-heurísticos multi-objetivos e suas hibridações.

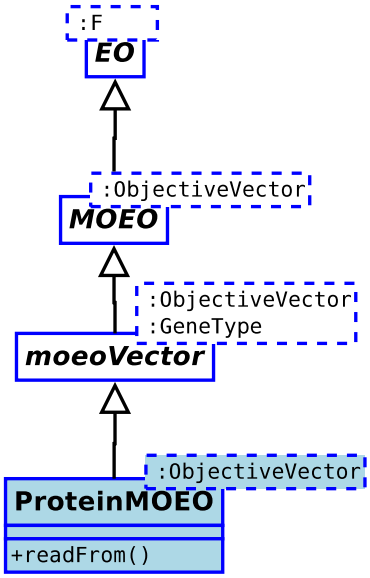


Figura 5.11: Diagrama UML da classe ProteinMOEO a qual representa a solução (Proteína). Esta é derivada da classe MOEO provida pelo componente ParadisEO-MOEO do ParadisEO. Assim, é possível visualizar informações a cerca da organização interna do ParadisEO no que tange a sua representação da solução.

4. **ParadisEO-PEO**: auxilia na paralelização e distribuição dos algoritmos meta-heurísticos.

O protpred-PEO-GROMACS utilizou-se do componente ParadisEO-MOEO, o qual permite incorporar vários AEMOs difundidos na literatura.

A integração entre o protpred-GROMACS com o ParadisEO foi o desenvolvimento de uma camada *wrapper*[1] que usa a estrutura de dados e as funções já desenvolvidas no protpred-GROMACS. Desta forma, visando facilitar o entendimento da implementação da camada *wrapper* foi produzido diagramas UML (*Unified Modeling Language*), os quais permitem aos desenvolvedores uma visualização lógica do desenvolvimento.

A representação da solução foi baseada derivando a classe MOEO provida pelo componente ParadisEO-MOEO do ParadisEO. Esta nova classe, chamada ProteinMOEO, é um ponteiro para a estrutura de dados definida no ProtPred-GROMACS, a qual manipula a solução (proteína). A Figura 5.11 ilustra o diagrama UML da classe ProteinMOEO.

Esta compacta implementação evidencia uma das características mais atraentes do *framework* ParadisEO: sua facilidade de incorporar as estruturas de dados e as funções já desenvolvidas em outro projeto com um mínimo esforço de codificação; para tanto, é necessário somente um *wrapper*. Assim, torna-se possível integrar o protpred-GROMACS em um grande número de algoritmos meta-heurísticos os quais já se encontram disponíveis no ParadisEO.

[1]Refere-se a capacidade de interação entre projetos distintos, porém possuindo a necessidade de interação, ou seja, compartilhando funcionalidades.

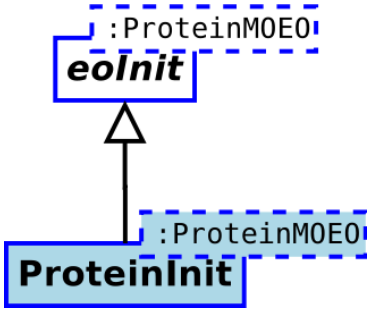


Figura 5.12: Representação do diagrama UML da classe ProteinInit que herda a classe eoInit do ParadisEO. Esta nova classe tem a finalidade de incorporar ao ParadisEO a população inicial. Os detalhes da criação da população inicial encontram-se na Subseção 5.1.

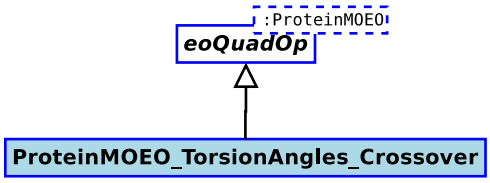


Figura 5.13: Diagrama UML da classe ProteinMOEO_TorsionAngles_Crossover a qual foi herdada da classe eoQuadOp do ParadisEO. Esta nova classe representa a integração do operador genético crossover do protpred-GROMACS no ParadisEO.

Neste sentido, as classes foram customizadas, respectivamente, da inicialização, dos operadores genéticos, da avaliação do *fitness* e dos resultados.

A população inicial encontra-se na classe ProteinInit que herda a classe eoInit do ParadisEO. A Figura 5.12 mostra a representação do diagrama UML desta nova classe. A criação da população inicial é finalidade do sistema protpred-GROMACS_PopInit. Assim, esta nova classe tem o objetivo de incorporar ao ParadisEO a população inicial. Os detalhes da criação da população inicial encontram-se na Subseção 5.1.

Os operadores genéticos do protpred-GROMACS foram mapeados nas classes ProteinMOEO_TorsionAngles_Crossover e ProteinMOEO_TorsionAngles_Mutation, respectivamente, crossover e mutação. A Figura 5.13 mostra a classe ProteinMOEO_TorsionAngles_Crossover a qual foi herdada da classe eoQuadOp do ParadisEO. Esta nova classe representa a integração do operador genético crossover do protpred-GROMACS no ParadisEO.

Já a Figura 5.14 ilustra a classe ProteinMOEO_TorsionAngles_Mutation a qual foi herdada da classe eoMonOp do ParadisEO. A classe herdada representa a integração do operador genético de mutação do protpred-GROMACS no ParadisEO.

Com relação a obtenção dos *fitness* foi desenvolvida a classe ProteinMOEOPopEval. Nesta classe ocorre a computação dos objetivos conforme a integração do algoritmo ProtPred-GROMACS (ver Seção 5.1). Assim, esta classe foi herdada da classe eoPopLoopEval que por sua vez herda da classe eoPopEvalFunc. Estas duas últimas classes pertencem ao *framework* ParadisEO. A primeira classe enfatiza a população corrente que necessita ser avaliada. Assim, para cada solução (conformação) é executada a computa-

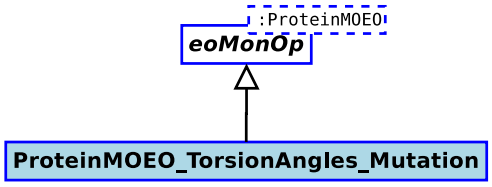


Figura 5.14: Diagrama UML da classe ProteinMOEO_TorsionAngles_Mutation a qual foi herdada da classe eoMonOp do ParadisEO. A classe herdada representa a integração do operador genético de mutação do protpred-GROMACS no ParadisEO.

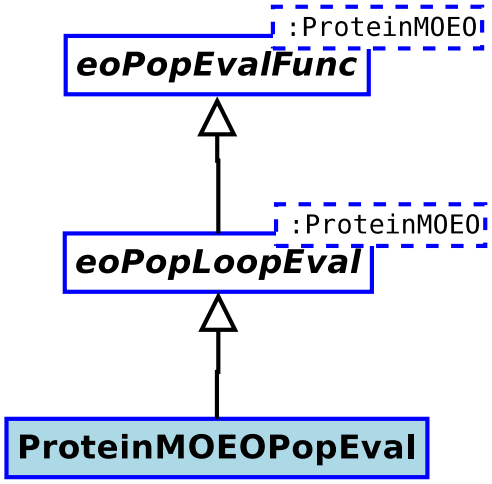


Figura 5.15: Representação do diagrama UML da computação dos *fitness* realizado pelo protpred-PEO-GROMACS. A classe ProteinMOEOPopEval enfatiza a integração do algoritmo protpred com o GROMACS. Assim, esta classe herda a classe eoPopLoopEval que por sua vez herda da classe eoPopEvalFunc. Estas duas últimas classes pertencem ao *framework* ParadisEO.

ção do seu *fitness*. A Figura 5.15 representa o diagrama UML da computação dos *fitness* realizada pelo protpred-PEO-GROMACS.

5.5 Aplicação do ProtPred-PEO-GROMACS no PSP

A fim de demonstrar a capacidade do 3PG no que tange predizer a estrutura terciária de uma proteína conhecendo-se somente a sua sequência primária, foi então implementado dois algoritmos evolutivos multi-objetivo: NSGA-II e SPEA2. A escolha desses algoritmos é justificável, uma vez que os mesmos se encontram já disponíveis no ParadisEO. Agora, o critério para comparar as predições realizadas é utilizado a métrica RMSD (*Root Mean Square Deviation*).

Então, o emprego do *framework* proposto é dividido em duas etapas, as quais são destacadas logo abaixo:

1. **Estratégia de Busca**: Empregando os algoritmos NSGA-II e SPEA2 visa encon-

trar as melhores soluções não-dominadas. Nesta etapa, é possível conhecer o menor C-α RMSD de todas as soluções não-dominadas em todas as gerações, assim como o da última geração, as quais são conhecidas, respectivamente, melhor RMSD não-refinado. e final RMSD não-refinado.

2. **Refinamento Estrutural**: Por meio da aplicação da técnica de DM (Dinâmica Molecular) em cada um dos indivíduos da fronteira de Pareto final da última geração tem-se um refinamento estrutural. Em analogia ao passo anterior, o menor RMSD C-α obtido durante todas as simulações e o último frame são chamados, respectivamente, melhor RMSD refinado e final RMSD refinado.

A performance dos algoritmos NSGA-II e SPEA2 são avaliados em termos da qualidade da fronteira de Pareto e os RMSD dos indivíduos não dominados em relação da estrutura nativa em cada geração. Em ambos, a taxa de crossover de um-ponto é 0.4 e o valor da taxa de mutação baseia-se em uma média para mutar um único aminoácido por indivíduo. A população é de 200 indivíduos. Para o SPEA2, a sua população externa possui também o tamanho de 200 indivíduos. Para a técnica de dinâmica molecular, em cada indivíduo da fronteira de Pareto final foi executada a 1 ns com 0.5 fs *timestep*.

Em termos da estratégia de busca é pretendido mimetizar o processo de *folding*. A discussão na Seção 2.7 sobre esse processo pode ter um entendimento em que o determinante no processo de *folding* é um colapso hidrofóbico onde é envolvido propriedades tanto energética quanto estrutural. Assim sendo, neste projeto o referido colapso, é representado por meio de sete propriedades:

1. Energia Potencial da Proteína (Pot);
2. Área de acessibilidade ao solvente Apolar (aSASA);
3. Área de acessibilidade ao solvente Polar (pSASA);
4. Número de ligações intra-proteína de Hidrogênios (HB);
5. Raio de Giro da Proteína (RG);
6. Energia de Van der Waals (VdW);
7. Energia de Coulomb (Coul).

Desta forma, o processo de *folding* pode ser sumarizado em um cenário conflitante no qual é necessário minimizar as propriedades Pot, aSASA, RG, VdW Coul e, simultaneamente, maximizar HB e pSASA. Para este cenário conflitante justifica-se o uso do multi-objetivo, conforme discorrido com mais ênfase na Seção 3.1. Então, cada uma das propriedades da proteína em termos da POMO são conhecidas como objetivos.

Um outro aspecto em que torna o processo de *folding* ser multi-objetivo é a não possibilidade de utilizar pesos teóricos em cada um dos objetivos. Logo, cada um dos objetivos (sete ao todo) foram combinados em dois a dois, resultando-se em seis diferentes cenários conflitantes. Por conseguinte, em cada uma dessas combinações, o espaço de busca é explorado de acordo com suas contribuições a fim de encontrar a predição da estrutura 3D da nativa.

Para uma melhor apresentação dos resultados, as três predições realizadas estão separadas em seções. Na Seção 5.5.1 é demonstrado os resultados da predição do peptídeo 1PLW. A predição da proteína 1UAO é discorrida na Seção 5.5.2. Por fim, na Seção 5.5.3 é mostrado a predição da proteína 1VII.

5.5.1 Experimentos com 1PLW

A predição do peptídeo 1PLW pode ser analisada por meio da visualização das fronteiras de Pareto finais de cada uma das combinações dos objetivos, conforme ilustrado na Figura 5.16, na qual é possível salientar:

1. Para a combinação de objetivos Pot&aSASA, os dois algoritmos compartilham algumas soluções (Fig. 5.16a);

2. O objetivo VdW não é um fator estrutural determinante já que gerou as mesmas soluções em ambos algoritmos, conforme Figura 5.16f;

3. Por sua vez, as combinações de objetivos aSASA&HB, aSASA&RG or HB&RG, respectivamente representadas nas Figuras 5.16b, 5.16d e 5.16e, convergiram para apenas uma única solução na fronteira de Pareto final.

Na primeira etapa da predição, ou seja, a predição não-refinada, a acurácia de predição dos algoritmos NSGA-II e SPEA2 é demonstrada por meio do RMSD do C-α com a estrutura nativa. O menor (0.50Å) e o último não-refinado (1.40Å) para ambos algoritmos foi obtido com a combinação de objetivos Pot&aSASA do NSGA-II. Na Tabela 5.4 é apresentado os valores dos RMSD. Para este caso é interessante notar que o valor do RMSD do melhor não-refinado para o NSGA-II é muito semelhante para as três combinações de objetivos, respectivamente, Pot&aSASA do SPEA2, aSASA&HB e aSASA&pSASA.

Na segunda etapa, é então aplicado um refinamento estrutural por meio de simulações de Dinâmica Molecular em cada uma das soluções da fronteira de Pareto final. Em termos de RMSD com a nativa, é possível realizar uma comparação com a primeira etapa. De maneira geral, com o emprego da técnica de DM foi, possível encontrar RMSD menor. É importante ressaltar que essa melhora de RMSD é em virtude da capacidade dos AEMOs encontrar modelos fisicamente significativos. Assim, a técnica de refinamento foi capaz de caminhar em direção à estrutura nativa. Na Tabela 5.5, em sua primeira linha, indica os melhores valores de RMSD para cada uma das combinações de objetivos. Já a segunda

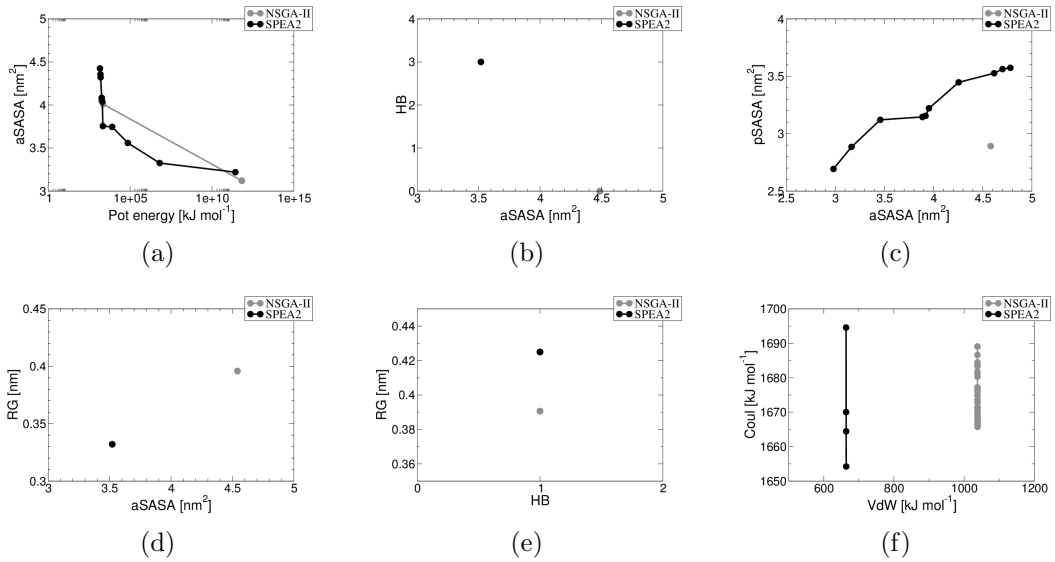


Figura 5.16: Representação das fronteiras de Pareto finais referentes à predição do peptídeo 1PLW pelos algoritmos NSGA-II e SPEA2 em várias combinações de objetivos.

Tabela 5.4: Valores dos RMSDs obtidos na predição do peptídeo 1PLW na etapa de exploração do espaço de busca. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD não-refinado e a segunda linha é o melhor RMSD final não-refinado. Todos os valores estão em Å.

	NSGA-II	SPEA2
Pot & aSASA	0.50	0.53
	1.40	0.97
aSASA & HB	0.53	0.63
	1.71	1.89
aSASA & pSASA	0.49	0.63
	1.83	1.44
aSASA & RG	0.61	0.88
	1.82	1.62
RG & HB	0.61	0.94
	1.56	1.78
VdW & Coul	0.67	0.64
	2.03	1.90

linha representa os melhores valores de RMSD final. É então possível concluir que o menor RMSD foi 0.60Å e o melhor final foi 0.79Å, ambos ocorridos na combinação de objetivos Pot & aSASA do algoritmo SPEA2.

Após a segunda etapa, a predição do peptídeo 1PLW pelo 3PG encontra-se concluída, o que permite uma comparação com outras técnicas de predições já difundidas na literatura. No trabalho de (CUTELLO; NARZISI; NICOSIA, 2006b) em que se aplicou o algoritmo PAES (KNOWLES; CORNE, 1999), tendo-se como operadores genéticos um sistema immune chamado de clonal, o RMSD foi 2.83Å. Em um outra predição, uma

Tabela 5.5: Valores dos RMSDs obtidos na predição do peptídeo 1PLW na etapa de refinamento estrutural. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD refinado e a segunda linha é o melhor RMSD final refinado. Todos os valores estão em Å.

	NSGA-II	SPEA2
Pot & aSASA	0.70	0.60
	1.82	0.79
aSASA & HB	0.71	0.82
	1.45	1.77
aSASA & pSASA	0.64	0.59
	1.00	1.22
aSASA & RG	1.04	0.90
	2.10	1.54
RG & HB	0.84	0.62
	1.54	1.07
VdW & Coul	0.97	0.60
	1.37	1.59

implementação paralela do NSGA-II, na qual foi aplicada em dois casos variando a quantidade de objetivos, ou seja, com 2 e 3, obteve-se RMSD *all-atom*, respectivamente, 2.65 e 1.89Å.

Na Tabela 5.6 está sumarizada os RMSDs da predição do 3PG em cada etapa, assim como cada um dos trabalhos de predições difundidos na literatura. É possível observar que a predição realizada pelo *framework* proposto em termos de RMSD ou foi melhor ou bem próxima.

Tabela 5.6: Melhores RMSDs em termos C-α, *backbone* e *all atom* para o peptídeo 1PLW. Todos os valores estão em Å.

	C-α	Backbone	All atoms
3PG - Etapa I (não-refinado)	0.50	0.79	2.60
3PG - Etapa II (refinado)	0.60	0.45	1.92
Literatura - PAES-Clonal	0.49	-	2.83
Literatura - NSGA-II-2	-	-	2.65
Literatura - NSGA-II-3	-	-	1.89

A Figura 5.17 mostra as imagens das melhores estruturas em termos de RMSDs. Conforme já evidenciado pelo RMSD *all-atom*, a estrutura refinada tem um melhor efeito sobre a cadeia lateral do que sobre o *backbone*. Desta forma, é provada a capacidade do SPEA2 em conseguir estrutura com significância física e, então, no refinamento, a habilidade de caminhar em direção à estrutura nativa.

(a) Nativa. (b) Melhor RMSD não-refinado. (c) Melhor RMSD refinado.

Figura 5.17: Representação das images das estruturas do peptídeo 1PLW, respectivamente: nativa, melhor RMSD não-refinado (Etapa I) e melhor RMSD refinado (Etapa II).

5.5.2 Experimentos com 1UAO

A predição do peptídeo 1UAO é analisada por meio da visualização das fronteiras de Pareto finais de cada uma das combinações dos objetivos, conforme ilustrado na Figura 5.18, sendo então possível evidenciar:

1. Para a combinação de objetivos Pot&aSASA, uma solução do algoritmo NSGA-II domina uma solução do SPEA2 (Fig. 5.18a);

2. Na combinação de objetivos aSASA&pSASA, algumas soluções do NSGA-II domina algumas soluções do SPEA2. Entretanto, não é possível afirmar que tais soluções são as melhores soluções, já que a fronteira de Pareto do NSGA-II encontra-se em uma região com altos valores tanto de aSASA quanto de pSASA, conforme Figura 5.18c.

3. Tendo o mesmo comportamento da predição anterior, o objetivo vdw não conseguiu distinguir soluções, conforme Figura 5.18f.

Na primeira etapa da predição, a acurácia dos algoritmos NSGA-II e SPEA2 são demonstradas por meio do RMSD do C-α com a estrutura nativa. A Tabela 5.7 idendifica os RMSD em cada combinação de objetivos para cada um dos algoritmos. Um comportamento interessante é que, diferentemente da predição anterior, os valores do RMSD não se originaram da mesma combinação de objetivos. Este fato ilustra o aumento da complexidade do espaço de busca, já que o número de resíduo dobrou-se. Assim sendo, o melhor RMSD foi de 1.07Å na combinação aSASA&RG. Já o melhor RMSD final, cujo valor é 2.76Å, originou-se da combinação aSASA&HB.

Na etapa de refinamento (segunda etapa), nota-se na Tabela 5.8 que as soluções providas pelo SPEA2 são mais fisicamente viáveis do que as soluções oriundas do NSGA-II. Em

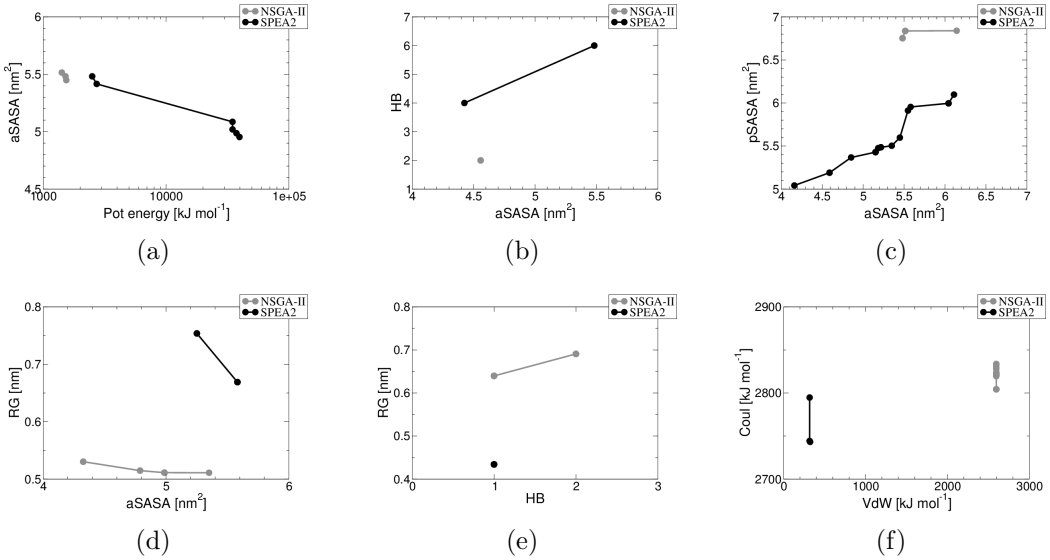


Figura 5.18: Representação das fronteiras de Pareto finais referentes à predição do peptídeo 1UAO pelos algoritmos NSGA-II e SPEA2 em várias combinações de objetivos.

Tabela 5.7: Valores dos RMSDs obtidos na predição do peptídeo 1UAO na etapa de exploração do espaço de busca. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD não-refinado e a segunda linha é o melhor RMSD final não-refinado. Todos os valores estão em Å.

	NSGA-II	SPEA2
Pot & aSASA	2.02	1.70
	3.60	4.06
aSASA & HB	1.50	1.55
	2.65	2.76
aSASA & pSASA	1.27	2.20
	3.47	2.82
aSASA & RG	2.01	1.07
	3.62	5.21
RG&HB	1.29	1.61
	4.86	3.04
VdW&Coul	1.70	2.49
	3.14	5.07

outras palavras, foi possível obter um menor RMSD, com estrutura nativa da 1UAO, com as soluções refinadas a partir da fronteira de Pareto final fornecido pelo algoritmo SPEA2. Na combinação de objetivo aSASA&pSASA salienta uma observação no que tange o valor do RMSD das soluções oriundas dos algoritmos. O valor foi de 2.07Å, porém, com o emprego da Dinâmica Molecular, foi possível encontrar conformações abaixo desse valor (1.77Å). O menor RMSD obtido entre todas as simulações de refinamento foi de 1.57Å, o qual foi atribuído pela combinação Pot&aSASA.

Tabela 5.8: Valores dos RMSDs obtidos na predição do peptídeo 1UAO na etapa de refinamento estrutural. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD refinado e a segunda linha é o melhor RMSD final refinado. Todos os valores estão em Å.

	NSGA-II	SPEA2
Pot & aSASA	1.73 3.25	1.57 2.24
aSASA & HB	2.20 3.02	1.63 2.57
aSASA & pSASA	2.07 2.75	1.77 2.07
aSASA & RG	1.86 3.99	1.76 2.61
RG&HB	2.38 4.18	2.16 2.85
VdW&Coul	2.71 3.76	2.65 4.24

Na literatura há dois trabalhos referentes à predição da 1UAO. No primeiro, (MAUPETIT; DERREUMAUX; TUFFÉRY, 2010) utilizou o algoritmo PEP-FOLD por meio da metodologia de *coarse-grained fragment assembly* empregando a análise de *clusters* mais populado (MPC[2]) e melhor cluster (BC[3]), os quais obtiveram um RMSD *all-atom* de 3.5 e 2.0Å, respectivamente. Diante destes valores, o 3PG atingiu uma melhor performance no que tange à predição desta proteína frente a predição MPC do PEP-FOLD. Por outro lado, com uma pequena diferença, menos de 0.50, PEP-FOLD pelo cluster BC obteve melhor predição. Já no segundo trabalho de predição da 1UAO o qual é o estado da arte, SEIBERT et al. (2005), empregando a técnica *Replica Exchange Molecular Dynamics* (REMD), obteve um RMSD *all-atom* de 1.8Å. É visto que, embora o desempenho do 3PG não conseguiu superá-lo, assim como o PEP-FOLD, o mesmo aproximou-se. A sumarização das comparações das predições encontra-se na Tabela 5.9.

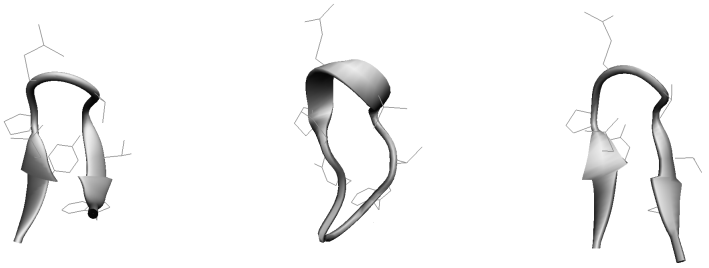
Tabela 5.9: Melhores RMSDs em termos C-α, *backbone* e *all atom* para o peptídeo 1UAO. Todos os valores estão em Å.

	C-α	Backbone	All atoms
3PG - Etapa I (não-refinado)	1.07	1.04	2.41
3PG - Etapa II (refinado)	1.57	1.18	2.19
Literatura - REMD	0.98	-	1.8
Literatura - PEP-FOLD	-	-	BC: 2.0 , MPC: 3.5

É possível avaliar a predição da 1UAO pelo 3PG não somente por meio de valores de RMSD, mas também de forma visual. A Figura 5.19 ilustra a estrutura nativa (Fig.

[2]do inglês most populated cluster.
[3]do inglês best cluster.

5.19a) da 1UAO, juntamente com as preditas, respectivamente, na Etapa I e Etapa II. Na Figura 5.19b representa a melhor predição da Etapa I em que há o surgimento de uma α-hélice. Por um outro lado, com o uso do refinamento, Etapa II, foi possível conseguir a estrutura β-*turn* conforme Figura 5.19c. É plausível ressaltar que os valores as estruturas obtidas em cada etapa não representam a mesma combinação de objetivos, mas somente a estratégia de busca. Assim, salienta-se a eficácia do SPEA2 em encontrar, num espaço de busca complexo, estruturas fisicamente flexível a ponto de, aplicando a DM, a mesma se direciona para a estrutura nativa.



(a) Nativa. (b) Etapa I. (c) Etapa II.

Figura 5.19: Ilustração da comparação da estrutura nativa e cada uma das etapas da predição pelo 3PG do peptídeo 1UAO.

5.5.3 Experimentos com 1VII

A predição da proteína 1VII é analisada por meio da visualização das fronteiras de Pareto finais de cada uma das combinações dos objetivos, conforme Figura 5.20. É possível comentar que:

1. Em ambas as combinações aSASA&HB e VdW&Coul o algoritmo NSGA-II tem menos diversidade, no entanto, visualmente domina o SPEA2 (ver Figuras 5.20b e 5.20f)

2. Já na combinação dos objetivos aSASA&pSASA, Figura 5.20c, um interessante comportamento das fronteiras finais dos algoritmos SPEA2 e NSGA-II nas quais há, praticamente, uma sobreposição.

Na primeira etapa da predição da proteína 1VII, a acurácia dos algoritmos NSGA-II e SPEA2 são demonstradas por meio do RMSD do C-α com a estrutura nativa. A Tabela 5.10 resume tais valores. Embora o fato da combinação dos objetivos aSASA&pSASA para os algoritmos terem próximas suas fronteiras, o SPEA2 teve uma melhor performance, já que o mesmo possui um menor RMSD final. O melhor não-refinado RMSD é atribuído

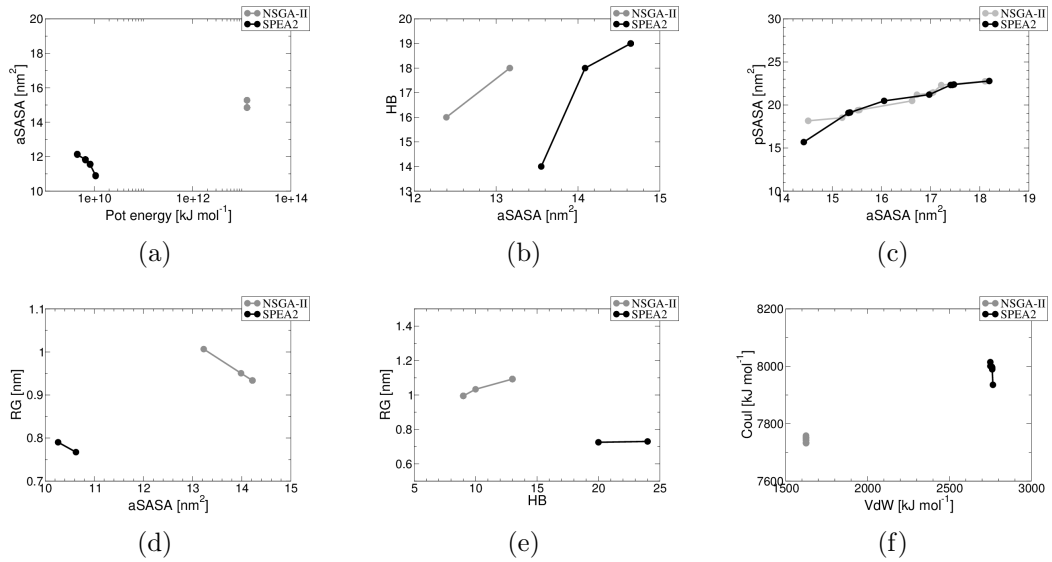


Figura 5.20: Representação das fronteiras de Pareto finais referentes à predição da proteína 1VII pelos algoritmos NSGA-II e SPEA2 em várias combinações de objetivos.

Tabela 5.10: Valores dos RMSDs obtidos na predição da proteína 1VII na etapa de exploração do espaço de busca. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD não-refinado e a segunda linha é o melhor RMSD final não-refinado. Todos os valores estão em Å.

	NSGA-II	SPEA2
Pot & aSASA	5.43	5.43
	8.66	9.00
aSASA & HB	5.87	6.30
	7.41	9.34
aSASA & pSASA	5.37	5.84
	9.02	6.95
aSASA & RG	5.86	5.24
	7.15	8.49
RG&HB	5.43	5.75
	6.21	6.93
VdW&Coul	5.41	5.40
	6.26	6.63

à combinação de objetivos aSASA&RG (5.24Å). Já a combinação RG&GB do NSGA-II com RMSD final de 6.21Å é o melhor RMSD final. Diferentemente das predições anteriores, esta proteína, sendo a maior, evidencia seu complexo espaço de busca, uma vez que o melhor e o final valor de RMSD foi obtido não somente por combinação de objetivos diferentes, mas também por algoritmos diferentes.

Na etapa de refinamento, nota-se na Tabela 5.11 que os RMSDs finais são melhores quando comparados com seu respectivo da etapa anterior, exceto o objetivo HB&RG. O melhor RMSD refinado foi 4.06Å pela combinação VdW&Coul. A diferença da acurácia

Tabela 5.11: Valores dos RMSDs obtidos na predição da proteína 1VII na etapa de refinamento estrutural. Para cada combinação de objetivo, a primeira linha indica o melhor RMSD refinado e a segunda linha é o melhor RMSD final refinado. Todos os valores estão em Å.

	NSGA-II	SPEA2
Pot & aSASA	5.66 7.31	6.80 7.48
aSASA & HB	5.78 6.51	5.43 6.43
aSASA & pSASA	6.09 7.03	4.39 4.56
aSASA & RG	5.50 6.56	6.95 7.97
RG & HB	5.56 5.87	6.54 8.00
VdW & Coul	5.17 5.44	4.06 4.96

de predição dos algoritmos com a combinação aSASA&pSASA é mantida após o refinamento e, além disso, nesta combinação tem-se o melhor RMSD final. Por fim, conforme nas predições anteriores, o SPEA2 gerou soluções com uma maior relevância física.

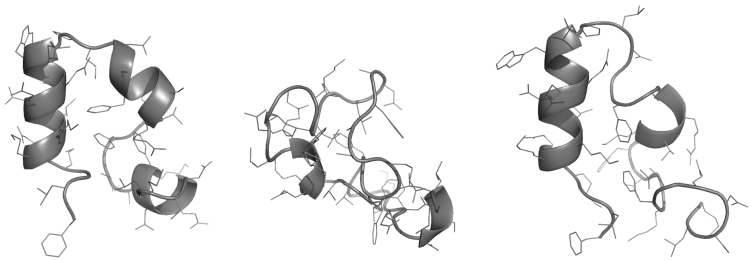
Na literatura existem dois trabalhos referentes à predição da proteína 1VII. O ZAGROVIC et al. (2002) fornece o melhor valor absoluto de predição da 1VII. Neste trabalho foi aplicado a técnica de Dinâmica Molecular em que, em virtude da sua necessidade computacional, empregou-se computadores distribuídos pelo mundo. Este projeto chama-se folding@home. A acurácia de predição foi RMSD C-α de 3.6 $\pm$ 1.3Å. Já o segundo, MAUPETIT; DERREUMAUX; TUFFÉRY (2010), foi utilizado o algoritmo PEP-FOLD em que por meio da análise de *clusters* mais populado (MPC) e melhor *cluster* (BC) obtiveram um RMSD *all-atom* de 7.4Å e 4.7Å, respectivamente. A sumarização das predições está na Tabela 5.12 na qual é possível visualizar que as predições realizadas pelo 3PG estão entre os melhores valores difundidos na literatura.

Tabela 5.12: Melhores RMSDs em termos C-α, *backbone* e *all atom* para a proteína 1VII. Todos os valores estão em Å.

	C-α	Backbone	All atoms
3PG - Etapa I (não-refinado)	5.24	4.92	6.49
3PG - Etapa II (refinado)	4.06	3.73	5.57
Literatura - Folding@home	3.6 $\pm$ 1.3	-	-
Literatura - PEP-FOLD	-	-	BC: 4.7 MPC:7.4

A predição da proteína 1VII pode ser também analisada de forma visual. Na Figura 5.21 apresenta a estrutura nativa da proteína 1VII (ver Figura 5.21a). Já a Figura 5.21b apresenta a melhor predição da Etapa I, na qual o algoritmo SPEA2 explorando o espaço

de busca em que somente conhecendo a sua sequência primária e tendo os objetivos conflitantes aSASA&RG, conseguiu encontrar a topologia nativa. Aplicando a técnica de DM para realizar o refinamento das predições realizadas pelos AEMOs, embora não destacado na classificação da Etapa I em termos de melhor e final RMSD, as soluções da fronteira providas pelo SPEA2 com a combinação VdW&Coul atingiu um melhor refinamento, sendo então assumida como a melhor predição do 3PG para a proteína 1VII. Esta predição encontra-se ilustrada na Figura 5.21c, onde é possível visualizar que a topologia nativa foi mantida e, com o refinamento, a predição aproximou da estrutura nativa.



(a) Nativa. (b) Etapa I. (c) Etapa II.

Figura 5.21: Ilustração da comparação da estrutura nativa e cada uma das etapas da predição pelo 3PG da proteína 1VII.

Capítulo 6

Conclusões

Este trabalho propõe o desenvolvimento do ProtPred-PEO-GROMACS, ou 3PG. Este consiste de um *framework* de computação evolutiva multi-objetivo para a predição *ab initio* da estrutura terciária de proteínas. Em linhas gerais, tal *framework* originou-se da integração entre os projetos: GROMACS, ParadisEO e ProtPred. O primeiro é um *framework* para modelagem molecular, enquanto que o segundo, é um *framework* de computação evolutiva. Já o ProtPred contém a aplicação de algoritmos evolutivos na predição da estrutura terciária de proteínas.

Cada projeto citado já é difundido na literatura em suas áreas. Por meio do ProtPred torna-se possível integrar ambos os projetos, sendo que dessa integração é que resultou o desenvolvimento do 3PG.

Em virtude da complexidade da integração proposta, a mesma fora dividida em duas etapas. A primeira, caracteriza-se pelo desenvolvimento do ProtPred-GROMACS, ou seja, o precursor do 3PG. A finalidade desta etapa é a implementação de um algoritmo evolutivo mono-objetivo em que a computação da função de aptidão das soluções seja realizada pelo GROMACS. Logo, tem-se a integração do ProtPred com o GROMACS.

Já a segunda etapa enfatiza-se o desenvolvimento do ProtPred-PEO-GROMACS onde o ProtPred-GROMACS integrou-se com o ParadisEO. Nesta etapa, o 3PG não somente prediz a estrutura terciária da proteína como também permite elucidar um ambiente de estudo das predições. Além disso, o 3PG mostrou ser capaz de prover um ambiente para o desenvolvimento, testes e comparações de desempenho de novos algoritmos evolutivos, aplicando-os no problema proposto.

Tais características puderam ser comprovadas por meio do emprego dos algoritmos NSGA-II e SPEA2. Estes algoritmos já se encotram disponibilizados no ParadisEO no que tange as suas características gerais. Porém, eles foram adequados à predição por meio de *wrappers* com o ProtPred-GROMACS. A qualidade das predições realizadas por cada algoritmo foram refinadas, por meio da técnica de Dinâmica Molecular provida pelo GROMACS. Com esta metodologia de predição, foi possível comparar as predições do 3PG com outras técnicas: Algoritmos Evolutivo Multi-Objetivo, *Replica Exchange Molecular*

Dynamics, PEP-FOLD e Folding@Home.

Os resultados indicam que o algoritmo SPEA2 tem uma melhor performance do que o NSGA-II na predição da estrutura terciária de proteínas em termos de variabilidade, visualização da dominância de Pareto, acurácia e flexibilidade física das estruturas geradas, visto que este algoritmo utiliza uma população externa de indivíduos não dominados, a qual é atualizada em cada geração, a fim de produzir novos indivíduos. Por um outro lado, o NSGA-II emprega um conceito de rank para todos os indivíduos e, então, não há garantia de que os novos indivíduos serão mantidos pelo seu elitismo.

Além disso, o uso das simulações de Dinâmica Molecular é um meio eficiente de aumentar a acurácia de predições usando somente função baseada na física. Os resultados aqui apresentados indicam que o SPEA2 tem a capacidade de predizer uma topologia similiar à nativa, a qual é suficientemente flexisível para ser refinada pela simulação de Dinâmica Molecular. A intregação destas duas técnicas resultam em uma interessante metodologia em termos de equilíbrio entre custo computacional e a qualidade de predição, as quais não poderiam ser obtidas quando aplicando-as independentemente. Vale ressaltar que o desenvolvimento e a aplicação de métodos baseados em dinâmica molecular com o intuito de prover um refinamento na predição é um dos mais interessantes tópicos neste campo (YANG; ZHANG, 2009).

6.1 Trabalhos Futuros

O trabalho proposto não se esgota todas as possiblidades para a predição da estrutura terciária de proteínas. Sendo então, em virtude dessa necessidade de novas implementações, evidencia-se como próximos trabalhos a implementação paralela do 3PG, uma vez que o mesmo já estando integrado com o ParadisEO, torna-se então possível usufruir do seu módulo ParadisEO-PEO. Além disso, pode-se realizar a implementação do algoritmo MEAT (*Multi-objective Evolutionary Algorithm on Tables*) a fim de aumentar as possibilidades de exploração do espaço de busca deste problema.

Referências

AARTS, E.; KORST, J. **Simulated Annealing and Boltzmann Machines**: a stochastic approach to combinatorial optimization and neural computing. [S.l.]: John Wiley and Sons, 1989.

ALBA, E.; LUQUE, G.; GARCIA-NIETO, J.; ORDONEZ, G.; LEGUIZAMON, G. MALLBA: a software library to design efficient optimisation algorithms. **Int. J. Innov. Comput. Appl.**, Inderscience Publishers, Geneva, SWITZERLAND, v.1, p.74–85, April 2007.

ANFINSEN, C. B. Principles that govern the folding of protein chains. **Science**, [S.l.], v.181, n.96, p.223–230, 1973.

BACK, T.; FOGEL, D.; MICHALEWICZ, Z. **Handbook of Evolutionary Computation**. [S.l.]: Institute of Physics Publishing and Oxford University Press, 1997.

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

BAXEVANIS, A.; OUELLETTE, B. **Bioinformatics - A practical guide to the analysis of genes and proteins**. [S.l.]: Lawrence Erlbaum Associates Publishers, 2001.

BJELKMAR, P.; LARSSON, P.; CUENDET, M. A.; HESS, B.; LINDAHL, E. Implementation of the CHARMM Force Field in GROMACS: analysis of protein stability effects from correction maps, virtual interaction sites, and water models. **Journal of Chemical Theory and Computation**, [S.l.], v.6, n.2, p.459–466, Feb. 2010.

BLEULER, S.; LAUMANNS, M.; THIELE, L.; ZITZLER, E. PISA: a platform and programming language independent interface for search algorithms. In: CONFERENCE ON EVOLUTIONARY MULTI-CRITERION OPTIMIZATION (EMO 2003), 2003, Berlin. **Anais...** Springer, 2003. p.494–508. (LNCS, v.2632).

BOTTEMA, O.; ROTH, B. **Theoretical Kinematics**. 1979.

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

BRASIL, C. R. S.; DELBEM, A. C. B.; BONETTI, D. R. F. Investigating relevant aspects of MOEAs for protein structures prediction. In: Proceedings of the 13th annual conference on Genetic and evolutionary computation, 2011, New York, NY, USA. **Anais...** ACM, 2011. p.705–712. (GECCO '11, v.1).

BRITANNICA, E. **Vilfredo Pareto.** 2007.

CALVO, J. C.; ORTEGA, J.; ANGUITA, M.; URQUIZA, J. M.; FLORIDO, J. P. Protein Structure Prediction by Evolutionary Multi-objective Optimization: search space reduction by using rotamers. In: INTERNATIONAL WORK-CONFERENCE ON ARTIFICIAL NEURAL NETWORKS, 10., 2009. **Proceedings...** [S.l.: s.n.], 2009. v.5517, p.861–868.

CHOU, S. M.; LEE, T. S.; SHAO, Y. E.; CHEN, I. F. Mining the breast cancer pattern using artificial neural networks and multivariate adaptive regression splines. In: EXPERT SYSTEMS WITH APPLICATIONS, 2004. **Proceedings...** [S.l.: s.n.], 2004. v.27, p.133–142.

COELLO, C.; PULIDO, G. Multiobjective Optimization using a Micro-Genetic Algorithm. In: GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE, 2001. **Proceedings...** Morgan Kaufmann Publishers, 2001. p.274–281.

Coello Coello, C. A. Evolutionary multi-objective optimization: a historical view of the field. **Computational Intelligence Magazine, IEEE**, [S.l.], v.1, n.1, p.28–36, Feb. 2006.

COHEN, B.; PRESNELL, S.; COHEN, F. Origins of structural diversity within sequentially identical hexapeptides. **Protein Science**, [S.l.], v.2, p.2134–2145, 1993.

COHEN, J. Bioinformatics an introduction for computer scientists. **ACM Comput. Surv.**, New York, NY, USA, v.36, n.2, p.122–158, 2004.

COPELAND, R. **Methods for Protein Analysis - A pratical guide to laboratory protocols**. [S.l.]: M. Chapman e Hall, 1993.

CORNE, D.; JERRAM, N.; KNOWLES, J.; OATES, M. PESA-II: region-based selection in evolutionary multiobjective optimization. In: GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE, 2001. **Proceedings...** Morgan Kaufmann Publishers, 2001. p.283–290.

CORNE, D.; KNOWLES, J.; OATES, M. The Pareto Envelope-Based Selection Algorithm for Multi-objective Optimisation. In: INTERNATIONAL CONFERENCE ON PARALLEL PROBLEM SOLVING FROM NATURE, 6., 2000, London, UK. **Proceedings...** Springer-Verlag, 2000. p.839–848.

CREIGHTON, T. **Protein Folding**. [S.l.]: W. E. Freeman and Company, 1992.

CRIVELLI, S.; KREYLOS, O.; HAMANN, B.; MAX, N.; BETHEL, W. ProteinShop: a tool for interactive protein manipulation and steering. **Journal of Computer-Aided Molecular Design**, [S.l.], v.18, p.271–285, 2004.

CUI, Y.; CHEN, R.; WONG, W. Protein Folding Simulation With Genetic Algorithm and SuperSecondary Structure Constraints. **Proteins**, [S.l.], v.31, p.247–257, 1998.

CUTELLO, V.; NARZISI, G.; NICOSIA, G. A multi-objective evolutionary approach to the protein structure predicition problem. **J. R. Soc. Interface**, [S.l.], v.83, p.1–13, 2006.

CUTELLO, V.; NARZISI, G.; NICOSIA, G. A multi-objective evolutionary approach to the protein structure predicition problem. **J. R. Soc. Interface**, [S.l.], v.83, p.1–13, 2006.

CUTELLO, V.; NARZISI, G.; NICOSIA, G. Computational Studies of Peptide and Protein Structure Prediction Problems via Multiobjective Evolutionary Algorithms. In: KNOWLES, J.; CORNE, D.; DEB, K.; CHAIR, D. R. (Ed.). **Multiobjective Problem Solving from Nature**. [S.l.]: Springer Berlin Heidelberg, 2008. p.93–114. (Natural Computing Series).

CUTELLO, V.; NARZISI, G.; NICOSIA, G.; PAVONE, M. Clonal Selection Algorithms: a comparative case study using effective mutation potentials. In: **Artificial Immune Systems**. [S.l.]: Springer Berlin / Heidelberg, 2004. p.13–28. (Lecture Notes in Computer Science, v.3627).

DARWIN, C. **On the Origin of Species By Means of Natural Selection**. [S.l.]: Gramercy, 1859.

DAY, R. O.; LAMONT, G. B.; PACHTER, R. Protein Structure Prediction by Applying an Evolutionary Algorithm. **Parallel and Distributed Processing Symposium, International**, Los Alamitos, CA, USA, v.0, p.155a, 2003.

DEB, K. **Multi-Objective Optimization using Evolutionary Algorithms**. [S.l.]: John Wiley and Sons, 2001.

DEB, K.; AGRAWAL, S.; PRATAB, A.; MEYARIVAN, T. **A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization**:

NSGA-II. [S.l.]: Indian Institute of Technology, Kanpur, India, 2000. KanGAL report. (200001).

DEJONG, K. A. **Evolutionary Computation**. [S.l.]: The MIT Press, 2006.

DILL, K. A. A.; OZKAN, S. B. B.; WEIKL, T. R. R.; CHODERA, J. D. D.; VOELZ, V. A. A. The protein folding problem: when will it be solved? **Curr Opin Struct Biol**, Department of Pharmaceutical Chemistry, University of California, San Francisco, CA 94143, USA., June 2007.

DILL, K. A.; BROMBERG, S. **Molecular Driving Forces**: statistical thermodynamics in chemistry & biology. 1.ed. [S.l.]: Garland Science, 2002.

DILL, K. A.; OZKAN, S. B.; SHELL, M. S.; WEIKL, T. R. The Protein Folding Problem. **Annual Review of Biophysics**, [S.l.], v.37, n.1, p.289–316, 2008.

DOOLITTLE, R. **Of URFs and ORFs**: a primer on how to analyze derived amino acid sequences. [S.l.]: University Science Books, 1986.

DRENTH, J. **Principles of Protein X-ray Crystallography**. [S.l.]: Springer, 1994. 368p.

DURILLO, J. J.; NEBRO, A. J. jMetal: a java framework for multi-objective optimization. **Advances in Engineering Software**, [S.l.], v.42, p.760–771, 2011.

ECHENIQUE, P. Introduction to protein folding for physicists. **Contemporary Physics**, [S.l.], v.48, n.2, p.81–108, 2007.

EISENBERG, D.; MCLACHLAN, A. D. Solvation energy in protein folding and binding. **Nature**, [S.l.], v.319, p.199–203, jan 1986.

EISENHABER, F.; LIJNZAAD, P.; ARGOS, P.; SANDER, C.; SCHARF, M. The double cubic lattice method: efficient approaches to numerical integration of surface area and volume and to dot surface contouring of molecular assemblies. **Journal of Computational Chemistry**, [S.l.], v.16, n.3, p.273–284, 1995.

EMMERICH, M.; HOSENBERG, R. **TEA - A Toolbox for the Design of Parallel Evolutionary Algorithms in C++**. [S.l.]: University of Dortmund, Germany, 2001.

ESHELMAN, L.; SCHAFFER, J. Real-coded genetic algorithms and interval schemata. In: FOUNDATIONS OF GENETIC ALGORITHMS, 1993. **Proceedings...** [S.l.: s.n.], 1993. v.2, p.187–202.

EZZIANE, Z. Applications of artificial intelligence in bioinformatics: a review. In: EXPERT SYSTEM WITH APPLICATIONS, 2006. **Proceedings...** [S.l.: s.n.], 2006. v.30, p.2–10.

FACCIOLI, R. A.; SILVA, I. N. da; DELBEM, A. C. B.; BRANCINI, G. T. P.; CALIRI, A. ProtPred-Gromacs: evolutionay algorithm with gromacs for protein structure prediction. **BIOMAT 2011 International Symposium on Mathematical and Computational Biology**, [S.l.], p.1–12, 2011.

FACCIOLI, R.; SILVA, I. N. da; BORTOT, L.; DELBEM, A. A Mono-Objective Evolutionary Algorithm with GROMACS for Protein Structure Prediction in Structural and Energetic Contexts. **IEEE World Congress on Computational Intelligence**, [S.l.], p.1–8, 2012.

FOGEL, D. An Introduction to Simulated Evolutionary Computation. **IEEE Transactions on Neural Networks**, [S.l.], v.5, p.3–14, 1994.

FONSECA, C.; FLEMING, P. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: FIFTH INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 1993. **Proceedings...** Morgan Kauffman Publishers, 1993. p.416–423.

GABRIEL, P. H. R.; MELO, V. V. de; DELBEM, A. C. B. Algoritmos evolutivos e modelo HP para predição de estruturas de proteínas. **Sba Controle & Automação**, [S.l.], v.23, n.1, p.25–37, 2012.

GAGNE, C.; PARIZEAU, M.; DUBREUIL, M. Distributed BEAGLE: an environment for parallel and distributed evolutionary computations. In: PROCCEEDINGS OF THE 17TH ANNUAL INTERNATIONAL SYMPOSIUM ON HIGH PERFORMANCE COMPUTING SYSTEMS AND APPLICATIONS HPCS 2003, 2003. **Anais...** [S.l.: s.n.], 2003. p.201.

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

HAIMING, L.; GARY, G. Rank-Density-Based Multiobjective Genetic Algorithm and Benchmark Test Function Study. In: IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, 2003. **Proceedings...** [S.l.: s.n.], 2003. v.7, p.325–343.

HAJELA, P.; LIN, C. Y. Genetic search strategies in multicriterion optimal design. **Structural Optimization**, [S.l.], v.4, p.99–107, 1992.

HAMILTON, W. R. On the Geometrical Demonstration of some Theorems obtained by means of the Quaternion Analysis. In: ROYAL IRISH ACADEMY, 1853. **Proceedings...** [S.l.: s.n.], 1853. v.5, p.407–415.

HANDL, J.; KELL, D. B.; KNOWLES, J. Multiobjective Optimization in Bioinformatics and Computational Biology. **IEEE/ACM Transactions on Computational Biology and Bioinformatics**, Los Alamitos, CA, USA, v.4, n.2, p.279–292, 2007.

HANDL, J.; LOVELL, S. C.; KNOWLES, J. Investigations into the Effect of Multiobjectivization in Protein Structure Prediction. In: PARALLEL PROBLEM SOLVING FROM NATURE, 10., 2008, Berlin, Heidelberg. **Proceedings...** Springer-Verlag, 2008. p.702–711.

HESS, B.; KUTZNER, C.; SPOEL, D. van der; LINDAHL, E. 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, March 2008.

HILBERT, M.; BÖHM, G.; JAENICKE, R. Structural relationships of homologous proteins as a fundamental principle in homology modeling. **Proteins**, [S.l.], v.17, p.138–151, 1993.

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

HOLLAND, J. **Adaptation in natural and artificial systems**. [S.l.]: MIT Press, 1992.

HONDA, S.; YAMASAKI, K.; SAWADA, Y.; MORII, H. **10 Residue Folded Peptide Designed by Segment Statistics**. [S.l.]: Cell Press,, 2004. 1507–1518p. v.12, n.8.

HONIG, B. Protein folding: from the levinthal paradox to structure prediction. **Journal of molecular biology**, Department of Biochemistry and Molecular Biophysics, Columbia University, 630 West 168 St., New York, NY 10032, USA. bh6@columbia.edu, v.293, n.2, p.283–293, Oct. 1999.

HORN, J.; NAFPLIOTIS, N.; GOLDBERG, D. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. In: FIRST IEEE CONFERENCE ON EVOLUTIONARY COMPUTATION, 1994. **Proceedings...** IEEE Service Center, 1994. v.1, p.82–87.

IGEL, C.; GLASMACHERS, T.; HEIDRICH-MEISNER, V. Shark. **Journal of Machine Learning Research**, [S.l.], v.9, p.993–996, 2008.

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.

JUNIOR, C. S.; SASSON, S. **Biologia**. [S.l.]: Saraiva, 2003.

KABSCH, W.; SANDER, C. Dictionary of protein secondary structure: pattern recognition of hydrogen bonded and geometrical features. **Biopolymers**, [S.l.], v.22, p.2577–2637, 1983.

KACZANOWSKI, S.; ZIELENKIEWICZ, P. Why similar protein sequences encode similar three-dimensional structures? **Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)**, [S.l.], v.125, n.3, p.643–650, Mar. 2010.

KARPLUS, M.; SHAKHNOVICH, E. **Protein Folding**. [S.l.]: W. H. Freeman and Company, 1992. p.125–195.

KHIMASIA, M.; COVENEY, P. Protein structure prediction as a hard optimization problem: the genetic algorithm approach. **Molecular Simulation**, [S.l.], v.19, p.205–226, 1997.

KITA, H.; YABUMOTO, Y.; MORI, N.; NISHIKAWA, Y. Multi-Objective Optimization by Means of the Thermodynamical Genetic Algorithm. In: INTERNATIONAL CONFERENCE ON PARALLEL PROBLEM SOLVING FROM NATURE, 4., 1996, London, UK. **Proceedings...** Springer-Verlag, 1996. p.504–512.

KLEEMAN, M.; LAMONT, G. Solving the aircraft engine maintenance scheduling problem using a multi-objective evolutionary algorithm. In: EVOLUTIONARY MULTI-CRITERION OPTIMIZATION LECTURE NOTE IN COMPUTER SCIENCE, 2005. **Proceedings...** Springer-Verlag Berlin: Heidelberg Platz, 2005. v.3410, p.782–796.

KLEPEIS, J. L.; FLOUDAS, C. A. Floudas. 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**, [S.l.], v.85, p.2003, 2003.

KNOWLES, J.; CORNE, D. The Pareto Archived Evolution Strategy: a new baseline algorithm for pareto multiobjective optimisation. In: CONGRESS ON EVOLUTIONARY COMPUTATION, 1999. **Proceedings...** IEEE Press, 1999. v.1, p.98–105.

KOSLOVER, E. F.; WALES, D. J. Geometry optimization for peptides and proteins: comparison of cartesian and internal coordinates. **The Journal of chemical physics**, [S.l.], v.127, n.23, p.234105, 2007.

KRZYSZTOF; GINALSKI. Comparative modeling for protein structure prediction. **Current Opinion in Structural Biology**, [S.l.], v.16, n.2, p.172–177, 2006.

LAUMANNS, M.; G., R.; H., S. A Spatial Predator-Prey Approach to Multi-Objective Optimization: a preliminary study. In: PARALLEL PROBLEM SOLVING FROM NATURE, 1998. **Proceedings...** Springer, 1998. p.241–249.

LEACH, A. R. **Molecular Modelling - Principles and Applications**. [S.l.]: Pearson, 2001.

LEACH, A. R.; GILLET, V. J. **An Introduction to Chemoinformatics**. [S.l.]: Springer, 2007.

LEOPOLD, P. E.; MONTAL, M.; ONUCHIC, J. N. Protein folding funnels: a kinetic approach to the sequence-structure relationship. **Proceedings of the National Academy of Sciences of the United States of America**, [S.l.], v.89, n.18, p.8721–8725, September 1992.

LEVINTHAL, C. Are there pathways for protein folding? **Journal de Chimie Physique et de Physico-Chimie Biologique**, [S.l.], v.65, p.44–45, 1968.

LI, H.; TANG, C.; WINGREEN, N. S. Nature of Driving Force for Protein Folding: a result from analyzing the statistical potential. **Phys. Rev. Lett.**, [S.l.], v.79, p.765–768, Jul 1997.

LIEFOOGHE, A.; BASSEUR, M.; JOURDAN, L.; TALBI, E.-G. ParadisEO-MOEO: a framework for evolutionary multi-objective optimization. In: EVOLUTIONARY MULTI-CRITERION OPTIMIZATION, 2007. **Proceedings...** Springer-Verlag-Berlin, 2007. v.4403, p.386–400.

LIMA, T. W. de. **Algoritmos Evolutivos para Predição de Estruturas de Proteínas**. 2006. Dissertação (Mestrado em Engenharia Elétrica) — Instituto de Ciências Matemáticas e de Computação - ICMC-USP.

LIMA, T. W. de; FACCIOLI, R. A.; GABRIEL, P. H. R.; DELBEM, A. C. B.; SILVA, I. N. da. Evolutionary approach to protein structure prediction with hydrophobic interactions. In: GENETIC AND EVOLUTIONARY COMPUTATION, 9., 2007, New York, NY, USA. **Proceedings...** ACM, 2007. p.425–425.

LIU, X.; BAI, F.; OUYANG, S.; WANG, X.; LI, H.; JIANG, H. Cyndi: a multi-objective evolution algorithm based method for bioactive molecular conformational generation. **BMC Bioinformatics**, [S.l.], v.10, n.1, p.101, March 2009.

LODISH, H.; BERK, A.; MATSUDAIRA, P.; KAISER, C. A.; KRIEGER, M.; SCOTT, M. **Biologia Celular e Molecular**. [S.l.]: Artmed, 2004.

LUND, M.; TRULSSON, M.; PERSSON, B. Faunus: an object oriented framework for molecular simulation. **Source code for biology and medicine**, [S.l.], v.3, n.1, Feb. 2008.

MARCOTTE, I.; SEPAROVIC, F.; AUGER, M.; GAGNÉ, S. M. **A Multidimensional 1H NMR Investigation of the Conformation of Methionine-Enkephalin in Fast-Tumbling Bicelles**. [S.l.]: Cell Press, 2004. 1587–1600p. v.86, n.3.

MÁRQUEZ-CHAMORRO, A.; DIVINA, F.; AGUILAR-RUIZ, J.; BACARDIT, J.; ASENCIO-CORTÉS, G.; SANTIESTEBAN-TOCA, C. A NSGA-II Algorithm for the Residue-Residue Contact Prediction. In: GIACOBINI, M.; VANNESCHI, L.; BUSH, W. (Ed.). **Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics**. [S.l.]: Springer Berlin / Heidelberg, 2012. p.234–244. (Lecture Notes in Computer Science, v.7246).

MAUPETIT, J.; DERREUMAUX, P.; TUFFÉRY, P. A fast method for large-scale De Novo peptide and miniprotein structure prediction. **J. Comput. Chem.**, MTi, INSERM UMR-S973 and RPBS, Université Paris Diderot - Paris 7, 5 rue Marie-Andrée Lagroua Weill-Halle, 75205 Paris, Cedex 13, France; Laboratoire de Biochimie Théorique, UPR 9080 CNRS, Institut de Biologie Physico-Chimique and Université Paris Diderot - Paris 7, 13 rue Pierre et Marie Curie, 75005 Paris, France, v.31, n.4, p.726–738, 2010.

MCGARRAH, D.; JUDSON, R. Analysis of the genetic algorithm method of molecular conformation determination. **J. Comput. Chem.**, [S.l.], v.14, n.11, p.1385–1395, 1993.

MCKNIGHT, C. J.; MATSUDAIRA, P. T.; KIM, P. S. NMR structure of the 35-residue villin headpiece subdomain. **Nature Structural & Molecular Biology**, [S.l.], v.4, n.3, p.180–184, Mar. 1997.

MICHALEWICZ, Z. **Genetic algorithms + Data Structures = Evolution Programs**. [S.l.]: Springer-Verlag New York, Inc., 1996.

MICHALEWICZ, Z.; SCHOENAUER, M. Evolutionary Algorithms for Constrained Parameter Optimization Problems. **Evolutionary Computation**, [S.l.], v.4, p.1–32, 1996.

MOHAN, K. S.; SHEIK, S. S.; RAMESH, J.; BALAMURUGAN, B.; JEYASIMHAN, M.; MAYILARASI, C.; SEKAR, K. CADB-2.0: conformation angles database. **Acta Crystallographica Section D**, [S.l.], v.61, n.5, p.637–639, May 2005.

MOULT, J.; FIDELIS, K.; KRYSHTAFOVYCH, A.; ROST, B.; TRAMONTANO, A. Critical assessment of methods of protein structure prediction—Round VIII. **Proteins: Structure, Function, and Bioinformatics**, [S.l.], v.77, n.S9, p.1–4, 2009.

NAIR, N.; GOODMAN, J. M. Genetic Algorithms in Conformational Analysis. **Journal of Chemical Information and Computer Sciences**, [S.l.], v.38, n.2, p.317–320, March 1998.

ONUCHIC, J. N.; SOCCI, N.; LUTHEY-SCHULTEN, Z.; WOLYNES, P. G. Funnels: the nature of the transition state ensemble. **Folding Design**, [S.l.], v.1, p.441–450, 1996.

OPELLA, S.; MARASSI, F.; GESELL, J.; VALENTE, A.; KIM, Y.; OBLATT-MONTAL, M.; MONTAL, M. Structures of the M2 channel-lining segments from nicotinic acetylcholine and NMDA receptors by NMR spectroscopy. **Nat Struct Mol Biol**, [S.l.], v.6, n.4, p.374–379, 1999.

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

PARSONS, J.; HOLMES, J. B.; ROJAS, J. M.; TSAI, J.; STRAUSS, C. E. M. Practical conversion from torsion space to Cartesian space for in silico protein synthesis. **J. Comput. Chem.**, Texas Agricultural Experiment Station, Texas A&M University, College Station, Texas 77843, USA., v.26, n.10, p.1063–1068, 2005.

PEITSCH, M. About the use of protein models. **Bioinformatics**, [S.l.], v.18, p.934–938, 2002.

PENDHARKAR, P. C.; RODGER, J. A.; YAVERBAUM, G. J.; HERMAN, N.; BENNER, M. Association, statistical, mathematical and neural approaches for mining breast cancer patterns. In: EXPERT SYSTEMS WITH APPLICATIONS, 1999. **Proceedings...** [S.l.: s.n.], 1999. p.223–232.

PETSKO, G.; RINGE, D. **Proteins Structure and Function**. [S.l.]: New Science Press Ltd, 2004.

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

RAMACHANDRAN, G.; SASISKHARAN, V. Conformation of polypeptides and proteins. **Protein Chem.**, [S.l.], v.23, p.283–437, 1968.

RIPON, K.; SAM, S.; MAN, K. A real-coding jumping gene genetic algorithm (RJGGA) for multiobjective optimization. **Information Sciences**, [S.l.], v.177, p.632–654, 2007.

RUDOLPH, G. Evolutionary Search under Partially Ordered Fitness Sets. In: INTERNATIONAL NAISO CONGRESS ON INFORMATION SCIENCE INNOVATIONS, 2001. **Proceedings...** ICSC Academic Press: Millet/Sliedrecht, 2001. p.818–822.

SAREL, J.; LEAVER-FAY, A.; CORN, J. E.; STRAUCH, E.-M.; KHARE, S. D.; KOGA, N.; ASHWORTH, J.; MURPHY, P.; RICHTER, F.; LEMMON, G.; MEILER, J.; BAKER, D. F. RosettaScripts: a scripting language interface to the rosetta macromolecular modeling suite. **PLoS ONE**, [S.l.], v.6, n.6, p.e20161, 06 2011.

SCHAFFER, J. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: FIRST INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 1985. **Proceedings...** Lawrence Erlbaum, 1985. p.93–100.

SCHAFFER, J. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: FIRST INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 1985. **Proceedings...** Lawrence Erlbaum, 1985. p.93–100.

SCHULZ, G.; SCHIRMER, R. **Principles of Protein Structure**. [S.l.]: Springer-Verlag, 1979.

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.

SEIBERT, M. M.; PATRIKSSON, A.; HESS, B.; SPOEL, D. van der. Reproducible Polypeptide Folding and Structure Prediction using Molecular Dynamics Simulations. **Journal of Molecular Biology**, [S.l.], v.354, n.1, p.173–183, 2005.

SILVERMAN, B. W. **Density Estimation for Statistics and Data Analysis**. [S.l.]: London:Chapman and Hall, 1986.

SIMONS, K. T.; BONNEAU, R.; RUCZINSKI, I.; BAKER, D. Ab initio protein structure prediction of CASP III targets using ROSETTA. **Proteins**, Department of Biochemistry, University of Washington, Seattle 98195, USA., v.Suppl 3, p.171–176, 1999.

SPOEL, D. van der. **Structure and dynamics of peptides**: theoretical aspects of protein folding. 1996. Tese (Doutorado em Engenharia Elétrica) — PhD Thesis, University of Groningen.

SPOEL, D. van der; HESS, B. GROMACS: the road ahead. **Wiley Interdisciplinary Reviews: Computational Molecular Science**, [S.l.], v.1, n.5, p.710–715, 2011.

SPOEL, D. van der; LINDAHL, E.; HESS, B.; BUUREN, A. R. van; APOL, E.; MEULENHOFF, P. J.; TIELEMAN, D. P.; SIJBERS, A. L. T. M.; FEENSTRA, K. A.; DRUNEN, R. van; BERENDSEN, H. J. C. **Gromacs User Manual version 4.0**. [S.l.]: Gromacs, 2009. Manual. (1–330).

SRINIVAS, N.; DEB, K. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. **Evolutionary Computation**, [S.l.], v.2, n.3, p.221–248, Fall 1994.

SUBRAMANI, A.; WEI, Y.; FLOUDAS, C. A. ASTRO-FOLD 2.0: an enhanced framework for protein structure prediction. **AIChE Journal**, [S.l.], v.58, n.5, p.1619–1637, 2012.

SUH, C. H.; RADCLIFFE, C. W. **Kinematics and mechanisms design**. [S.l.]: Wiley, New York, 1978. 434p.

TALBI, E.-G. **Metaheuristics** : from design to implementation. [S.l.]: John Wiley & Sons, 2009.

TICONA, W. G. C. **Aplicação de Algoritmos Genéticos Multiobjetivos para Alinhamento de Sequências Biológicas**. 2003. Dissertação (Mestrado em Engenharia Elétrica) — Instituto de Ciências Matemáticas e de Computação - ICMC-USP.

TUFFERY; P.; ETCHEBEST; C.; HAZOUT; S.; LAVERY, R. A new approach to the rapid determination of protein side chain conformations. **J Biomol Struct Dyn**, [S.l.], v.8, n.6, p.1267–1289, 1991.

VELDHUIZEN, D. **Multiobjective Evolutionary Algorithms**: Classifications, Analyses, and New Innovations. 1999. Tese (Doutorado em Engenharia Elétrica) — Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio.

VULLO, A. On the Role of Machine Learning in Protein Structure Determination. **Journal of the Italian Association for Artificial Intelligence**, [S.l.], v.3, p.22–30, 2002.

WANGSHU, Y.; CHEN, S.; CHEN, Z. SDMOGA: a new multi-objective genetic algorithm based on objective space divided. In: INTERNATIONAL CONFERENCE ON NEURAL INFORMATION PROCESSING, 13., 2006. **Proceedings...** [S.l.: s.n.], 2006. v.3, p.754–762.

WENG, X.; HAMEL, L.; MARTIN, L.; PECKHAM, J. A genetic algorithm for energy minimization in bio-molecular systems. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION, 2005. **Proceedings...** [S.l.: s.n.], 2005. v.1, p.49–56.

WILSON, I.; HAFT, D.; GETZOFF, E.; TAINER, J.; LERNER, R.; BRENNER, S. Identical short peptide sequences in unrelated proteins can have different conformations: a testing ground for theories of immune recognition. **Proc. Natl. Acad. Sci.**, [S.l.], v.82, p.5255–5259, 1985.

YANG; ZHANG. Protein structure prediction: when is it useful? **Current Opinion in Structural Biology**, [S.l.], v.19, n.2, p.145–155, 2009.

ZAGROVIC, B.; SNOW, C. D.; SHIRTS, M. R.; PANDE, V. S. Simulation of Folding of a Small Alpha-helical Protein in Atomistic Detail using Worldwide-distributed Computing. **Journal of Molecular Biology**, [S.l.], v.323, n.5, p.927–937, 2002.

ZHANG, Y. I-TASSER server for protein 3D structure prediction. **BMC Bioinformatics**, [S.l.], v.9, n.1, p.40, 2008.

ZHANG, Y. Progress and challenges in protein structure prediction. **Current Opinion in Structural Biology**, [S.l.], v.18, n.3, p.342–348, June 2008.

ZITZLER, E.; DEB, K.; THIELE, L. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. **Evolutionary Computation**, [S.l.], v.8, n.2, p.173–195, 2000.

ZITZLER, E.; LAUMANNS, M.; THIELE, L. **SPEA2**: improving the Strength Pareto Evolutionary Algorithm. [S.l.]: Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, 2001. (103).

ZITZLER, E.; THIELE, L. **An Evolutionary Algorithm for Multiobjective Optimization**: The Strength Pareto Approach. [S.l.]: Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), 1998. (43).

Apêndices

Apêndice A

Cálculo das Propriedades da Proteína

Conforme já mencionado, o estudo das propriedades da proteína é por meio de suas conformações tridimensionais interagindo com o meio (ver Seção 2.8).

Diferentes conformações possuem diferentes energias. Tais energias são calculadas por uma variedade de técnicas computacionais, sendo as mais empregadas a mecânica quântica e a mecânica molecular. Para a quântica, a equação de Schrödinger (ou uma aproxinação da mesma) é então utilizada. Nesta equação envolvem tanto o núcleo quanto os elétrons da molécula e, portanto, exige um tempo computacional elevado. Já a mecânica molecular emprega uma representação mais simples onde somente a posição do núcleo é considerado (LEACH; GILLET, 2007).

Em virtude desta representação, para a mecânica molecular é utilizado os campos de forças. Eles por sua vez possuem as informações das contribuições do comprimento e dos ângulos de ligação, rotações dos ângulos torsionais, das interações eletrostática e van der Waals entre os átomos (LEACH; GILLET, 2007).

Neste trabalho foi empregado o *software* GROMACS com o campo de força CHARMM27 (BJELKMAR et al., 2010) para o cálculo das propriedades da proteína. Sendo assim, nas seções a seguir serão discorridos com mais ênfase as propriedades empregadas sob a ótica de suas implementações no GROMACS.

A.1 Energia Potencial

A energia potencial é calculada tendo as equações e os parâmetros do campo de força CHARMM27. Esta energia é o somatório de todas as interações covalentes e não-covalentes. Assim, a Equação (A.1) ilustra a referida energia.

$$E_{total} = E_{bonded} + E_{non-bonded} \tag{A.1}$$

onde o primeiro termo corresponde às interações covalentes enquanto que o segundo termo às não-covalentes.

As interações covalentes, representada na Equação (A.2), são modeladas como um somatório das seguintes: comprimento de ligação, ângulo de ligação, ângulo diedrais, diedrais impróprios e $1-3$ Urey-Bradley.

$$\begin{aligned} U_{bonded} = \sum_{bonds} k_b(b-b_0)^2 + \sum_{angles} k_\theta(\theta-\theta_0)^2 + \\ \sum_{dihedrals} k_\phi[1+cos(n\phi-\delta)] + \sum_{impropers} k_\omega(\omega-\omega_0)^2 + \\ \sum_{Urey-Bradley} k_u(u-u_0)^2 \end{aligned} \tag{A.2}$$

onde o primeiro termo refere-se ao energia de comprimento de ligação. Os parâmetros k_b é uma constante do campo de força, b é o comprimento de ligação e b_0 é o comprimento ideal. O segundo termo refere-se a energia do ângulo de ligação e seus parâmetros k_θ é o ângulo de força constante, θ é o valor do ângulo atual e θ_0 é o valor do ângulo ideal. A energia dos ângulos diedrais é o terceiro termo. Os parâmetros para este termo são: k_ϕ é o valor da força constante diedral, n é a multiplicidade, ϕ é o valor do ângulo atual e δ é o valor de mudança de fase. O quarto termo indica o valor da energia do ângulos diedrais impróprios, tais como desvios de planaridade em anéis aromáticos. Os parâmetros para o quarto termo são: k_ω é o valor da força imprópria, ω é o valor do ângulo atual e ω_0 é o valor ideal. O quinto e último termo, chamado potencial Urey-Bradley, representa um refinamento do ângulo considerando a distância de um átomo com o seu segundo vizinho. Os seus termos são: k_u é o valor constante de Urey-Bradley, u é o valor da distância atual e u_0 é a distância ideal (SPOEL et al., 2009).

Por sua vez, as interações não-covalentes da Equação (A.1) são ilustradas na Equação (A.3).

$$\begin{aligned} U_{non-bonded} = \sum_{i,j} \epsilon \left[\left(\frac{R_{i,j}}{r_{i,j}} \right)^{12} - \left(\frac{R_{i,j}}{r_{i,j}} \right)^{6} \right] \\ + \sum_{i,j} \frac{q_i q_j}{\epsilon r_{i,j}} \end{aligned} \tag{A.3}$$

onde as interações não-covalentes entre um par de átomos (i,j) são as contribuições de Van der Waals e Eletrostática. O primeiro termo é calculado por um potencial de 12-6 de Lennard-Jones, onde ϵ é o mínimo valor de energia da específica interação, $R_{i,j}$ é a distância onde o potencial é zero e $r_{i,j}$ é a distância atual entre i e j. Já o segundo termo é calculado com o potencial de Coulomb em que q_i e q_j são as cargas dos átomos i e j, respectivamente. Já ϵ é o parâmetro de permissidade no vácuo e $r_{i,j}$ é a distância entre os átomos i e j (SPOEL et al., 2009).

A.2 Raio de Giro

O raio de Giro é calculado por meio do programa g_gyrate, sendo que utiliza a seguinte equação (SPOEL et al., 2009):

$$RG = \left(\frac{\sum_i \|r_i\|^2 m_i}{\sum_i m_i} \right)^{\frac{1}{2}} \tag{A.4}$$

onde m_i representa a massa do átomo i e r_i é a posição do mesmo átomo com relação ao centro de massa da proteína.

A.3 Área da Superfície de Acessibilidade do Solvente

O cálculo da Área de Acessibilidade ao Solvente é realizada pelo GROMACS utilizando as equações descritas nos trabalhos de EISENBERG; MCLACHLAN (1986) e EISENHABER et al. (1995). Tais equações encontram-se implementadas no programa g_sas que por sua vez retorna as áreas, respectivamente, hidrofóbica, hidrofílica e total.

A.4 Ligações de Hidrogênios

As ligações de Hidrogênio são calculas pelo GROMACS empregando o programa g_hbond. Conforme já mencionado na Seção 2.8.5, essas ligações surgem quando o grupo doador possuindo um átomo de hidrogênio ligado à um elemento muito eletronegativo e o receptor possuindo um átomo muito eletronegativo. Assim, neste programa, por meio da determinação de *cutoffs* para os ângulos do doador e receptor, obtem-se o valor da ligação.