

Luis Paulo Barbour Scott

**Investigação da Utilização de Algoritmos
Genéticos e Redes Neurais Artificiais na
Predição de Estruturas Protéicas**



1910001217



Unesp-Universidade Estadual Paulista
IBILCE - Câmpus de São José do Rio Preto

Luis Paulo Barbour Scott

Luis Paulo Barbour Scott

Investigação da Utilização de Algoritmos
Genéticos e Redes Neurais Artificiais na
Predição de Estruturas Protéicas

**Investigação da Utilização de Algoritmos
Genéticos e Redes Neurais Artificiais na
Predição de Estruturas Protéicas**

Unesp- Universidade Estadual Paulista
IBILCE - Câmpus de São José do Rio Preto

Luis Paulo Barbour Scott

**Investigação da Utilização de Algoritmos
Genéticos e Redes Neurais Artificiais na
Predição de Estruturas Protéicas**

Tese de Doutorado em Biofísica
Molecular, área de concentração em
Biofísica Molecular, apresentada ao
Departamento de Física do Instituto de
Biotecnologia, Letras e Ciências Exatas da
Universidade Estadual Paulista "Júlio de
Mesquita Filho" Unesp.

Orientador: Prof. Dr. Jorge Chahine
Coorientador: Prof. Dr. José Roberto Ruggiero

São José do Rio Preto
Outubro de 2003



L. 1.217/03



7577.112
54252

20801017

Scott, Luis Paulo Barbour.

Investigação da utilização de redes neurais e algoritmos genéticos na
predição de estruturas proteicas / Luis Paulo Barbour Scott – São José
do Rio Preto : [s.n.], 2003

141 f. : il. ; 30 cm.

Orientador: Jorge Chahine

Tese (doutorado) – Universidade Estadual Paulista. Instituto de
Biociências, Letras e Ciências Exatas

1. Proteínas - Estrutura. 2. Bioinformática. 3. Algoritmos genéticos.
4. Redes neurais (Computação). 5. Predição - Estrutura. I. Chahine,
Jorge. II. Universidade Estadual Paulista. Instituto de Biociências,
Letras e Ciências Exatas. III. Título.

CDU – 577.112



Índice

Resumo.....	i
Abstract.....	ii
Listas de Siglas.....	iii
Capítulo 1-Introdução.....	01
1.1.Considerações Iniciais.....	01
1.2.Motivação do Trabalho.....	03
1.3.Objetivos do Trabalho.....	05
1.4.Organização da Monografia.....	05
Capítulo 2- Estruturas Protéicas e o Problema de Predição.....	07
2.1.Considerações Iniciais.....	07
2.2.Estruturas Protéicas.....	07
2.3.Alinhamento de Seqüências.....	12
2.4.Modelagem por Homologia.....	16
2.5.Predição 1D (Estrutura Secundária).....	17
2.6.Predição 2D e 3D (Estruturas Terciárias).....	23
2.7.Métodos Computacionais e o Problema de Predição.....	24

Capítulo 3- Redes Neurais Artificiais e a Predição de Estruturas Protéicas27

3.1.Considerações Iniciais.....	27
3.2.Conceitos Básicos.....	28
3.3.Tipos de Redes Neurais Artificiais (Exemplos de Rede).....	33
3.4.Regras de Aprendizagem	37

Capítulo 4- Algoritmos Genéticos e a Predição de Estruturas Protéicas.....40

4.1.Considerações Iniciais.....	41
4.2.Conceitos Básicos.....	42
4.3.Exemplo de Aplicação.....	46

Capítulo 5- Estudo do Núcleo Hidrofóbico do Citocromo_{b562}.....46

5.1.Considerações Iniciais.....	46
5.2.Biblioteca de Rotâmeros.....	47
5.3. Predição do Núcleo Hidrofóbico utilizando Redes de Hopfield.....	48
5.4. Predição do Núcleo Hidrofóbico utilizando Algoritmo Genético....	52

Capítulo 6- Algoritmos Genéticos Aplicados na Predição de Estrutura de Peptídeos.....61

6.1.Considerações Iniciais.....	61
6.2.Predição do Mastoporan-X.....	61

6.3. Predição da Polialanina e Poliisoleucina com a Cadeia Principal Relaxada.....	73
--	----

Capítulo 7- Predição 1D de Proteínas Usando Redes Neurais.....83

7.1. Considerações Iniciais.....	83
7.2. Bases de Dados.....	84
7.3. Interface do Pré-Processamento.....	86
7.4. Coeficientes.....	89
7.4. Simulação Utilizando um única rede neural.....	90
7.5. Implementação da Arquitetura com Duas Redes.....	92
7.6. Adicionando Propriedades Físico-Químicas dos Aminoácidos.....	97

Capítulo 8- Conclusões.....112

8.1. Introdução.....	112
8.2. Contribuições do Trabalho.....	126
8.3. Trabalhos Futuros.....	128

Referências Bibliográficas.....129

Bibliografia Consultada.....137

Anexo I.....139

Agradecimentos

A Deus, pelo amor e proteção.

Aos professores Jorge Chahim e José Roberto Angeleri, pela orientação e supervisão durante o desenvolvimento deste trabalho.

Aos meus pais e irmãos, pelo amor, carinho e apoio incondicional.

A todos os amigos e colegas que contribuíram com sugestões, comentários e apoio durante o desenvolvimento deste trabalho.

Aos meus familiares "Tia" e "Tio", pela compreensão e incentivo.

Ao meu amigo de infância, pelo apoio e incentivo durante o desenvolvimento deste trabalho.

Dedico esse Trabalho à Giselda pelo amor, paciência, carinho, cumplicidade e incentivo nos momentos mais difíceis.

Agradecimentos

À Fapesp, pelo apoio Financeiro

Aos professores Jorge Chahine e José Roberto Ruggiero, pela sábia orientação e pela confiança depositada em minha pessoa.

Aos meus pais e irmãos, pelo amor, incentivo e apoio incondicional.

A todos os amigos e colegas que diretamente ou indiretamente contribuíram para o desenvolvimento desse trabalho.

Aos meus cachorros “nina” e “fly”, pelo companheirismo incondicional.

Ao departamento de Física do Ibilce, pelo apoio ao desenvolvimento do trabalho.

Resumo

Esse trabalho consiste na aplicação de algoritmos genéticos, utilizando diferentes campos de força e diferentes bibliotecas de rotâmeros, no problema de predição e projeto de núcleos hidrofóbicos bem como na predição de estruturas de peptídeos. Também se utilizou rede neural (Hopfield) para realizar predição 3D de núcleos hidrofóbicos. Dois tipos de bibliotecas de rotâmeros foram utilizados: dependentes ou não da estrutura do backbone da proteína. Vários campos de força foram testados com crescente grau de complexidade, partindo-se de um campo no qual a única interação presente era do tipo Lennard-Jones, acrescentou-se o termo coulombiano e a seguir considerou-se os efeitos da solvatação, através de um termo de potencial proporcional a área acessível ao solvente. E, por fim, são projetadas e testadas diferentes arquiteturas de redes neurais do tipo MLP para realizar a predição 1D fazendo uso de diversas informações físico-químicas da proteína.



Abstract

This work uses genetic algorithms, distinct force fields and rotamer libraries to predict and to design hydrophobic cores. Besides, we used genetic algorithms to predict peptides structure. We also apply neural network (hopfield) to optimize and predict the structure of the protein hydrophobic core (Cytochrome b_{652}). In this work we use two kinds of rotamer libraries: dependent or not dependent of the protein backbone. Several force fields were tested with a growing complexity; we started it with a field where the only present interaction was Lennard-Jones. We added the coulombian term and we considered the solvation effects through a term proportional to the solvent accessible area. And finally, different architectures of Artificial Neural Networks (Multi Layers Perceptron) were projected and tested in the prediction of the secondary structures of proteins, taking as patterns the structures in helix form (H), beta sheet (E) and coil (C) and using several physical and chemical information of protein.



LISTA DE SIGLAS

AG- Algoritmo Genético

BLAST- Basic Local Alignment Search Tool.

DSSP- base de dados e método para converter coordenadas 3D em estrutura secundária.

MLP- Multi Layers Perceptron

PDB- Protein Data Bank.

PHD- simple profile-based neural network prediction

Predição 1D – Predição da Estrutura Secundária e ou Área Acessível ao Solvente.

Predição 3D – Predição da Estrutura Terciária.

RNAs- Redes Neurais Artificiais.



Introdução

1.1.Considerações Iniciais

Entre as várias classes de moléculas biológicas de grande importância para os seres vivos, encontram-se as proteínas. O termo proteína provém do grego (proteios) que significa “de primeira magnitude”. As proteínas são moléculas complexas que possuem uma estrutura terciária (tridimensional) específica. Estas macromoléculas realizam tarefas de extrema importância para o organismo, como a catálise de reações químicas, transporte, reconhecimento e transmissão de sinal. A função das proteínas conhecidas está determinada pela sua estrutura espacial. Dessa forma, obter ou conhecer a estrutura tridimensional da proteína no seu estado nativo, e compreender as forças que o estabilizam são dois problemas fundamentais da Biofísica de proteínas.

Conforme o paradoxo de Levinthal, uma pequena cadeia polipeptídica possui um número enorme de possíveis estados conformacionais (Chan & Dill, 1993) (Dill, 1990, 1995). Porém, a proteína alcança a sua forma nativa estável em uma escala de tempo infinitamente menor do que quando comparada à escala de tempo necessária para atingir o estado nativo caso a busca fosse ao acaso. Baseado neste fato, pode-se supor que, pelo menos “in vivo”, a busca não é aleatória, e sim dirigida por forças que conduzem a proteína para uma forma estável.

O balanço das forças responsáveis pelo enovelamento das proteínas em uma estrutura nativa pode estar na seqüência de aminoácidos da macromolécula e na sua interação com meio. A correspondência entre seqüência e estrutura é conhecida como “segundo código genético” (Dill, 1990, 1995).

Em princípio, esse conhecimento poderia permitir a projeção e desenvolvimento de novas proteínas (“*drug design*”) com aplicações abrangendo um vasto campo. Nos últimos anos, o novo “*design*” de proteínas cujo objetivo é nada mais nada menos, do que a projeção de seqüências protéicas artificiais para alcançar estruturas desejadas (terciárias e secundárias), têm sido tentado com um considerável esforço e sucesso. Em 1949, Anfinsen mostrou que a estrutura primária (ou seja, a seqüência de aminoácidos) de uma proteína pode determinar sua estrutura terciária. Dessa forma, existiria um conjunto de regras para dirigir o enovelamento da proteína a partir de sua estrutura primária até sua estrutura terciária. A busca por esse conjunto de regras pode ser conhecida como *problema do enovelamento protéico* (Dill, 1995).

Ao estudar o enovelamento de proteínas pretende-se descobrir quais as propriedades do polipeptídeo que levam a cadeia a adotar uma estrutura única e estável, e investigar como a seqüência de aminoácidos está relacionada com essas propriedades, e como ocorre o processo de enovelamento (Chan & Dill, 1993). A predição de estrutura, por sua vez, possui como objetivo investigar e propor ferramentas que permitem predizer a estrutura secundária e terciária de proteínas a partir de sua seqüência primária e de outras informações, ou seja procura uma descrição mais computacional.



As técnicas de predição de estrutura também podem auxiliar o estudo do problema de enovelamento de proteína. Métodos de predição de estrutura secundária fornecem informações que podem ser utilizadas tanto por algoritmos de predição *ab initio* bem como restrições (“condições de contorno”) adicionais para algoritmos de reconhecimento de enovelamento como também podem fornecer informações sobre as interações mais relevantes. A predição de estrutura secundária (1D) também pode ser utilizada como guia para estudos de *mutagenese* dirigida.

1.2.Motivação do Trabalho

A função de uma proteína está diretamente relacionada com a sua estrutura nativa, portanto é importante conhecermos a estrutura 3D dessas moléculas. O número de seqüências de proteínas (estruturas primárias) conhecidas e depositadas em bancos de dados está crescendo muito mais rápido do que a nossa habilidade de resolver as estruturas terciárias experimentalmente (Rost, 1998) (Rost, 1993). Portanto, técnicas eficazes para predição de estruturas são importantes para diminuir a diferença entre o número de seqüências depositadas e estruturas 3D determinadas.

Um método tradicionalmente, utilizado na predição de estruturas, é a modelagem por homologia. Porém, pode-se predizer a estrutura 3D de, aproximadamente, apenas 25% a 30% das seqüências de proteínas depositadas nos bancos de dados primários através desta técnica.

A Figura 1, apresenta dados referentes a dois bancos de dados: SWISS-PROT (<http://www.expasy.org/swissprot>) e YEST, indicando a porcentagem de seqüência depositadas que podem ter sua estrutura terciária definida por modelagem por homologia e a porcentagem para o qual isso não seria possível.



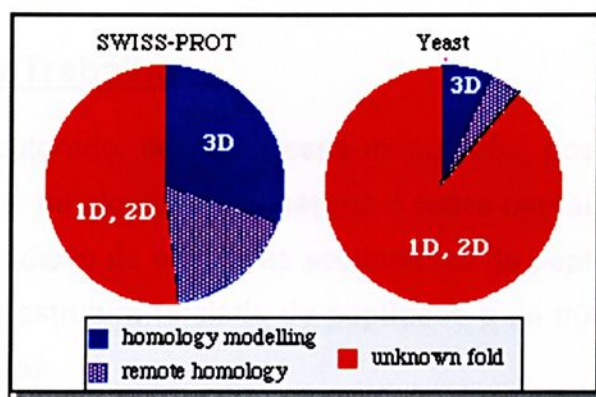


Figura 1- Ilustra, como exemplo, a quantidade de seqüências depositadas no SWISS-PROT e Yeast para o qual a estrutura terciária (3D) poderia ser obtida através de modelagem por homologia (Rost, 1999).

Não é ainda possível, através dos métodos existentes, prever com exatidão as estruturas tridimensionais (terciárias) de proteínas a partir apenas de sua seqüência primária (Rost & Sander, 1994).

Na última década, muito esforço vem sendo aplicado no desenvolvimento e na aplicação de novas metodologias e algoritmos para predição de estrutura secundária e terciária de proteínas. Vários métodos de simulação computacional como algoritmos genéticos e redes neurais artificiais vem sendo utilizados afim de prever estruturas secundárias e/ou terciárias (Rost, 1999) (Chandonia & Karplus, 1999) (Sun, 1995) (Dandekar & Argos, 1999). Métodos de simulação não estocásticos (como dinâmica molecular) também desempenham um papel importante no estudo conformacional de proteínas (Kollman, 2001) (Bohr et al., 1998) (Barton, 1995). O uso dessas técnicas e métodos em conjunto, explorando o ponto forte de cada um, vem sendo usado presentemente como uma alternativa para obter melhores predições de estrutura de proteínas (Rost & Sander, 1999).

1.3.Objetivos do Trabalho

O trabalho de doutorado, descrito nessa monografia, possui como objetivo investigar a utilização de algoritmos genéticos e redes neurais como processos de otimização na predição de estruturas secundárias de peptídeos e proteínas, bem como prever a estrutura terciária de peptídeos e de núcleos hidrofóbicos no interior de proteínas.

Dois ingredientes fundamentais na metodologia são: a biblioteca de rotâmeros para cadeias laterais dos aminoácidos e o potencial de interação entre os átomos. A metodologia faz uso de uma biblioteca de rotâmeros para as cadeias laterais dos resíduos de aminoácidos. Dois tipos de bibliotecas são investigadas: uma independente e outra dependente do *backbone* da proteína. Na primeira função potencial investigada, a interação entre os átomos consiste basicamente do termo do tipo Lennard-Jones. Após se certificar que os algoritmos eram eficientes para otimizar a busca conformacional, as interações entre átomos foram acrescidas do termo coulombiano e de um potencial de hidratação implícito, ou seja, proporcional à área acessível ao solvente.

Outro objetivo do trabalho é investigar como as características das proteínas, que constituem as bases de dados para treinar as redes neurais, e como os dados físico-químicos influenciam no desempenho da performance das mesmas para diferentes arquiteturas de redes neurais do tipo Multi Layer Perceptron (MLP) e Redes de Hopfield.

1.4.Organização da Tese

Essa Tese está organizada em 8 Capítulos. No Capítulo 1, é apresentada uma breve introdução ao assunto, a motivação do trabalho e os objetivos do mesmo. No Capítulo 2, faz-se uma discussão sobre estruturas de proteínas e o problema de predição de estruturas protéicas.

No Capítulo 3, discute-se, brevemente, sobre os principais conceitos de redes neurais artificiais e seus principais conceitos. No Capítulo 4, apresenta-se uma breve discussão sobre algoritmos genéticos e seus conceitos básicos e



aplicações na predição de estrutura protéica. No Capítulo 5, são apresentados os resultados da predição de núcleos hidrofóbicos através de algoritmos genéticos e redes neurais. No Capítulo seguinte, Capítulo 6, são descritos os resultados da predição de peptídeos utilizando algoritmos genéticos e biblioteca de rotâmeros. No Capítulo 7, é discutido o preditor 1D, desenvolvido como parte do trabalho de doutorado. Por último, no Capítulo 8, apresentam-se as conclusões, sugestões para trabalhos futuros e as contribuições do trabalho.

2.1. Considerações Iniciais

Nesta Capítulo são discutidos conceitos básicos sobre estrutura de proteínas e o problema da predição das mesmas. Na Seção 2.2, discutem-se conceitos básicos sobre estrutura de proteínas. Na Seção 2.3 apresentam-se os conceitos básicos sobre alinhamento de seqüências. A seguir, na Seção 2.4, é descrita a modelagem por homologia.

Na Seção 2.5 discute-se a Predição 1D, ou seja, a predição de estrutura secundária. Na Seção 2.6 são discutidas a Predição 2D e 3D. Na Seção 2.6 apresentam-se, brevemente, os métodos computacionais utilizados para predição 1D e 2D. Por fim, na Seção 2.7 são feitas algumas considerações finais.

2.2. Estruturas Protéicas

As proteínas são polímeros de unidades aminoácidas. Cada aminoácido contém um grupo amino (NH_2), um grupo ácido carboxílico (COOH) e um radical R ligados a um átomo de carbono, denominado carbono α (C_α) (Siryer, 1995) (Figura 2). Na natureza, existem principalmente sete tipos de aminoácidos que diferem pelo radical R e que compõem as proteínas. Estes

Capítulo 2

Estruturas Protéicas e o Problema de Predição

2.1.Considerações Iniciais

Nesse Capítulo são discutidos conceitos básicos sobre estrutura de proteínas e o problema de predição das mesmas. Na Seção 2.2, discutem-se conceitos básicos sobre estrutura de proteínas. Na Seção 2.3 apresentam-se os conceitos básicos sobre alinhamento de seqüências. A seguir, na Seção 2.4, é descrita a modelagem por homologia.

Na Seção 2.5 discute-se a Predição 1D, ou seja, a predição de estrutura secundária. Na Seção, 2.5 são discutidas a Predição 2D e 3D. Na Seção 2.6 apresentam-se, brevemente, os métodos computacionais utilizados para predição 1D e 3D. Por fim, na Seção 2.7 são feitas algumas considerações finais.

2.2.Estruturas Protéicas

As proteínas são polímeros de unidades aminoácidos. Cada aminoácido contém um grupo amina (NH_2), um grupo ácido carboxílico ($-\text{COOH}$) e um radical R ligados a um átomo de carbono, denominado carbono alfa (C_α) (Stryer, 1995) (Figura 2). Na natureza, existem principalmente vinte tipos de aminoácidos que diferem pelo radical R e que compõem as proteínas. Esse

aminoácidos podem ser classificados de acordo com as propriedades que sua cadeia lateral apresenta em: aminoácidos básicos, aminoácidos ácidos, aminoácidos polares e não polares (Stryer, 1995).

Portanto, pode-se definir uma proteína como uma cadeia de aminoácidos ligados covalentemente, sendo que esta ligação ocorre entre o grupo amina de um aminoácido e o grupo carboxila do aminoácido seguinte, formando a ligação peptídica (Figura 3) e pode-se visualizar uma proteína como um colar de contas.

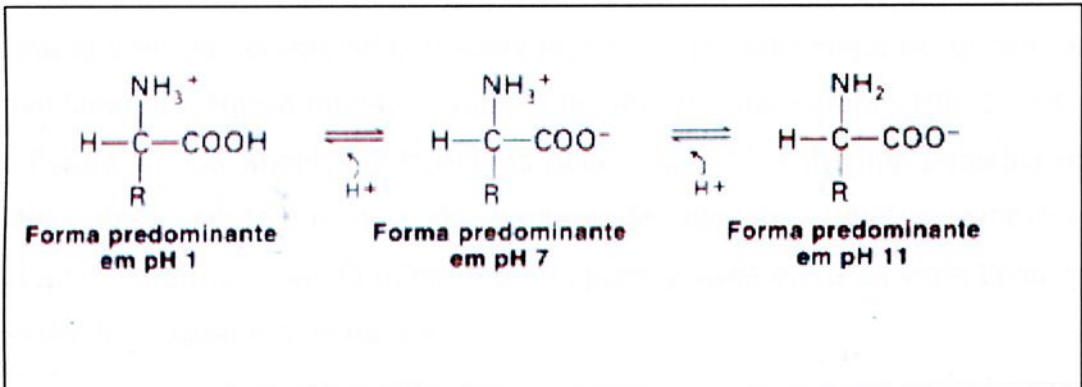


Figura 2- Ilustração de um aminoácido em suas três formas para diferente pH (Stryer, 1995).

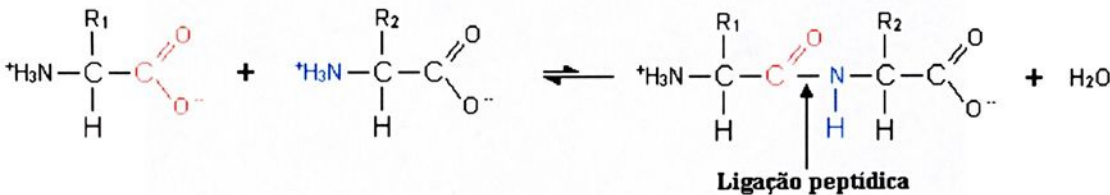


Figura 3- Ilustração de uma ligação peptídica (Stryer, 1995).

As ligações peptídicas possuem propriedades importantes para a estrutura de uma proteína. A ligação do carbono ao nitrogênio, na ligação peptídica, possui um caráter de ligação dupla parcial (ressonante) o que implica que a rotação em torno dessa ligação fica limitada a 180° aproximadamente para

cada peptídeo. Essa rigidez confere à ligação peptídica uma planaridade, assim a proteína poderia ser vista como uma sequência de planos conectados no carbono alfa. A liberdade rotacional nas ligações que conectam o carbono alfa (C_α) às duas unidades planares adjacentes formam dois ângulos chamados de Φ e Ψ , que conferem à cadeia principal da proteína, ou esqueleto (*backbone*), sua liberdade conformacional (Stryer, 1995).

Dessa forma pode-se descrever a estrutura da cadeia principal (*backbone*) de uma proteína através dos valores dos ângulos Φ e Ψ . Isto é, pode-se descrever a conformação de um resíduo na proteína a partir de um ponto em um mapa com as coordenadas desses ângulos chamado mapa ou gráfico de Ramachandran. Nesse mapa, os valores de Φ e Ψ variam entre -180° e $+180^\circ$ (ver Figura 4). Os ângulos χ formados pela ligação dos átomos pesados da cadeia lateral conferem a cada aminoácido também uma considerável liberdade conformacional. O número de ângulos χ varia entre os vinte tipos de aminoácidos existentes na natureza.

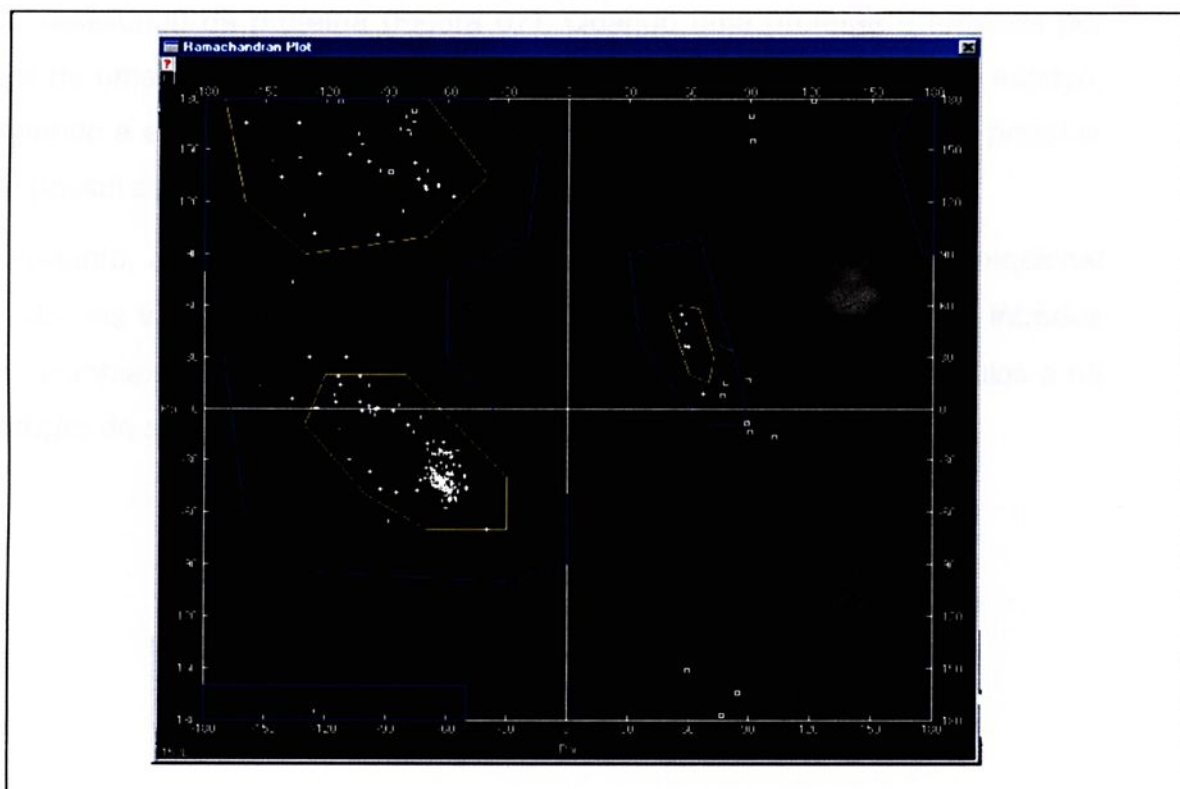


Figura 4- Ilustração de um gráfico de Ramachandran de uma proteína depositada no PDB (Protein Data Bank) (<http://www.rcsb.org/pdb>), gerado pelo software Pdbviewer. Os

pontos brancos indicam os valores dos ângulos para os aminoácidos de uma certa proteína.

A estrutura de uma proteína é, usualmente, hierarquizada em quatro níveis: estrutura primária, estrutura secundária, estrutura terciária e quaternária. A estrutura primária de uma proteína é o arranjo linear dos aminoácidos que constituem a cadeia polipeptídica, sendo que diferentes proteínas apresentam seqüências diferenciadas de aminoácidos (Stryer, 1995).

A estrutura secundária refere-se à organização local de partes da cadeia polipeptídica que podem arranjar-se de diferentes formas conhecidos como motivos estruturais. Tais motivos são determinados pelas combinações de ângulos Φ e Ψ . Os motivos estruturais mais conhecidos são as α -hélices (v. Figura 05), as β -folhas (v. Figura 06), voltas (*turns*) e fitas aleatórias (Stryer, 1995).

Uma única proteína pode exibir diferentes tipos de estrutura secundária que rearranjam-se espacialmente, enovelando-se, na chamada estrutura terciária (tridimensional) da proteína (Figura 07). Quando uma proteína é formada por mais de uma cadeia, suas estruturas terciárias podem arranjar-se, no espaço, formando a estrutura quaternária. Um exemplo bastante conhecido de proteína que possui estrutura quaternária é a Hemoglobina.

Portanto, as proteínas possuem um alto grau de liberdade conformacional devido aos valores que os ângulos Φ e Ψ e χ podem assumir e isso introduz uma complexidade na busca conformacional para essas macromoléculas e na predição de suas estruturas secundárias e terciárias (Stryer, 1995).

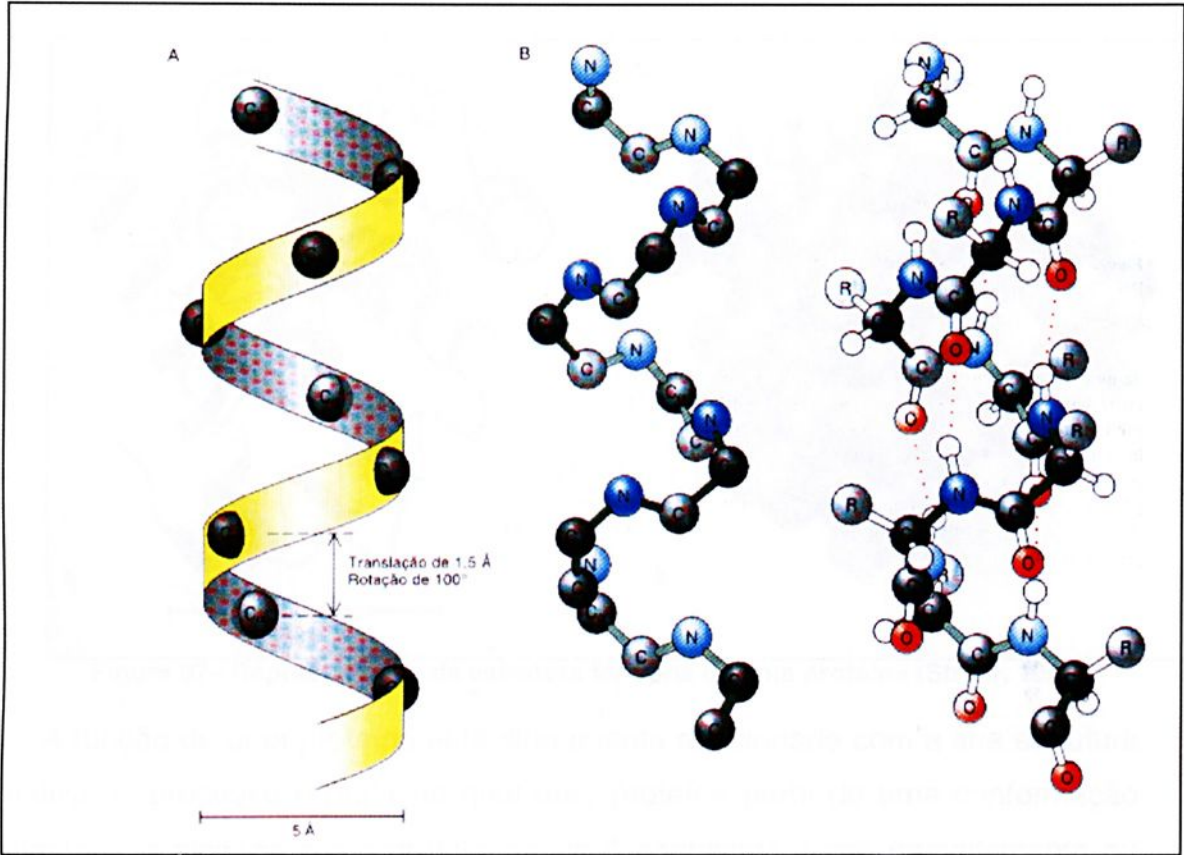


Figura 05- Ilustração de uma estrutura α -hélice (Stryer, 1995).

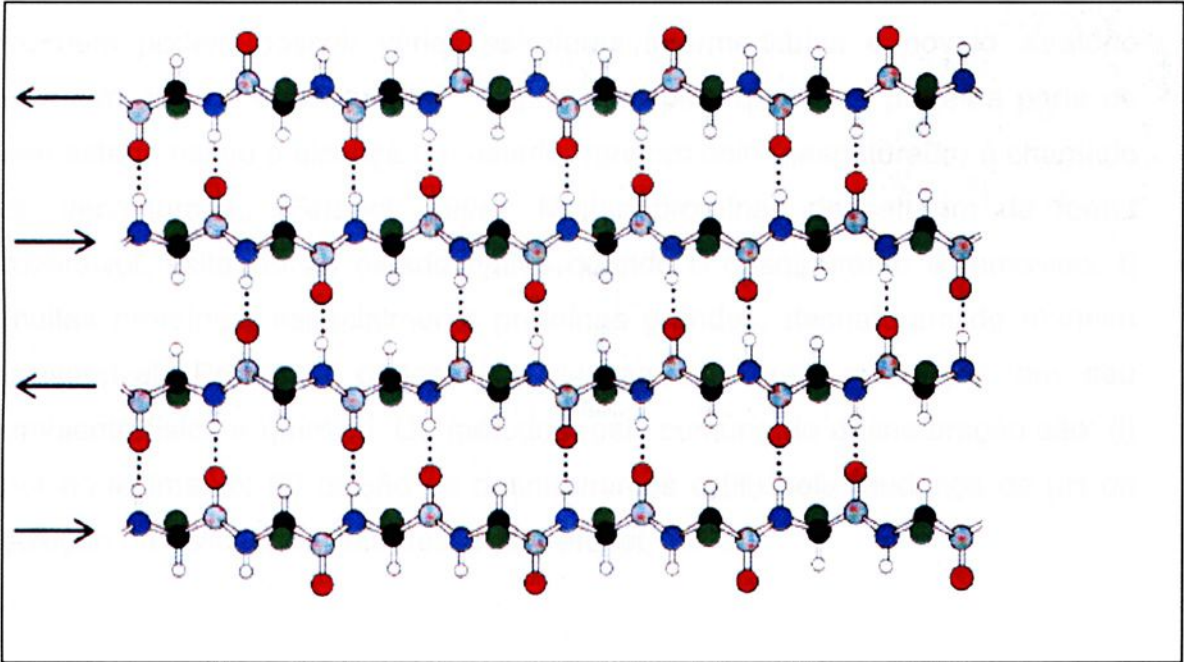


Figura 06- Ilustração de uma estrutura folha - β (Stryer, 1995).

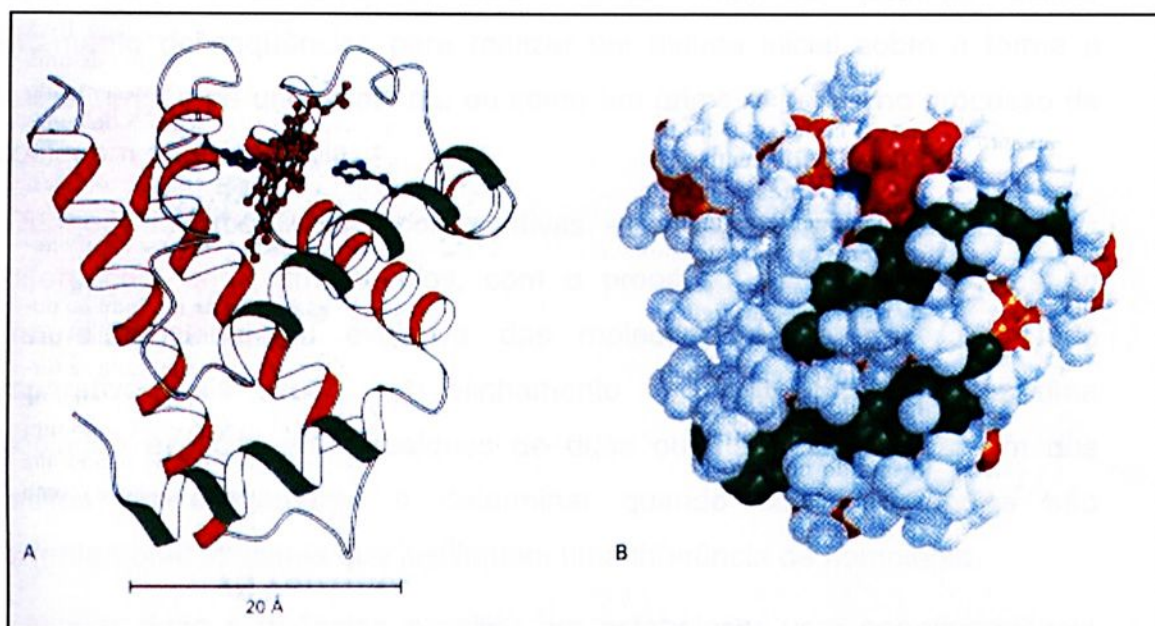


Figura 07– Representação da estrutura terciária de uma proteína (Stryer, 1995).

A função de uma proteína está diretamente relacionada com a sua estrutura nativa. O processo a partir do qual uma proteína parte de uma conformação aleatória e alcança sua estrutura nativa é conhecido como enovelamento ou *folding*.

Além de uma estrutura correspondente ao estado nativo, as proteínas também podem possuir várias estruturas intermediárias e novelo aleatório (*random coil*, ou desnaturada). O processo pelo qual uma proteína parte de seu estado nativo e alcança um estado “random coil” (desnaturado) é chamado de desnaturação (Fershet, 1998). Muitas proteínas desnaturam de forma reversível, voltando ao estado nativo quando o desnaturante é removido. E muitas proteínas, especialmente proteínas grandes, desnaturam de maneira irreversível. Proteínas podem ser desnaturadas por mudanças em seu ambiente físico e químico. Os métodos mais comuns de desnaturação são: (i) por aquecimento; (ii) adição de desnaturantes e (iii) pela mudança do pH da solução ou aplicando altas pressões (Fershet, 1998).

2.3. Alinhamento de Seqüências

Sabe-se que as proteínas que possuem uma alta identidade em relação à seqüência primária devem possuir estruturas 3D semelhantes, portanto, podem

possuir funções bastante parecidas. Dessa maneira, pode-se utilizar alinhamento de seqüências para realizar um estudo inicial sobre a forma e possível função de uma proteína, ou como um primeiro passo no processo de modelagem por homologia.

Quando fazemos análises comparativas, estamos procurando similaridades e diferenças, entre aminoácidos, com o propósito de inferir analogias na estrutura funcional ou evolutiva das moléculas envolvidas. O método comparativo mais usado é o alinhamento de seqüências, que faz uma associação explícita entre resíduos de duas ou mais seqüências. Um dos objetivos do alinhamento é determinar quando duas seqüências são suficientemente similares que justifiquem uma inferência de homologia.

Alinhar duas seqüências consiste em estabelecer uma correspondência entre os aminoácidos dessas seqüências, no caso de proteínas, de maneira que a ordem não seja violada. Por ordem entende-se que se os resíduos nas posições n_1 e n_2 ($n_1 < n_2$) estão associados aos resíduos m_1 e m_2 de outra seqüência, então m_1 deve ser menor que m_2 .

Não é necessário que todas as bases de uma seqüência estejam associadas às da outra seqüência, mas é desejável que o número de associações seja o maior possível (Vasconcelos, 2000). Uma maneira extremamente simples de visualização do alinhamento consiste na utilização de “gaps” (“buracos”) entre os elementos das seqüências de modo que o alinhamento ocorra verticalmente, como ilustrado na Figura 08. A Figura 09 ilustra um exemplo de alinhamento múltiplo realizado pelo software Clustal-X.

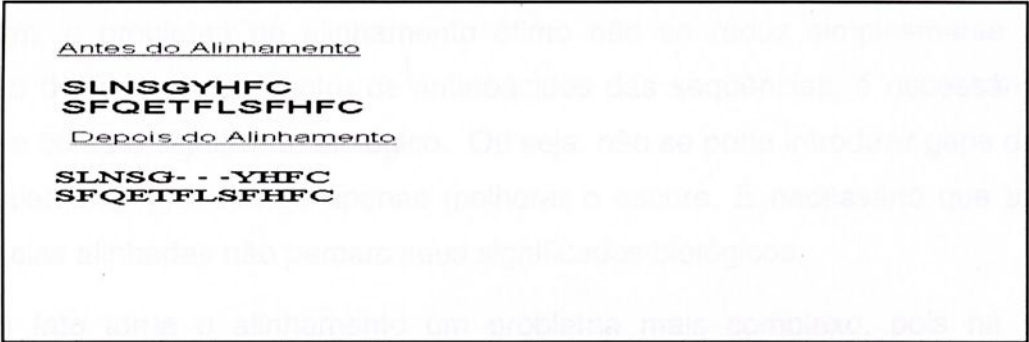


Figura 08– Exemplo de Alinhamento de duas Seqüências Hipotéticas de Aminoácidos.

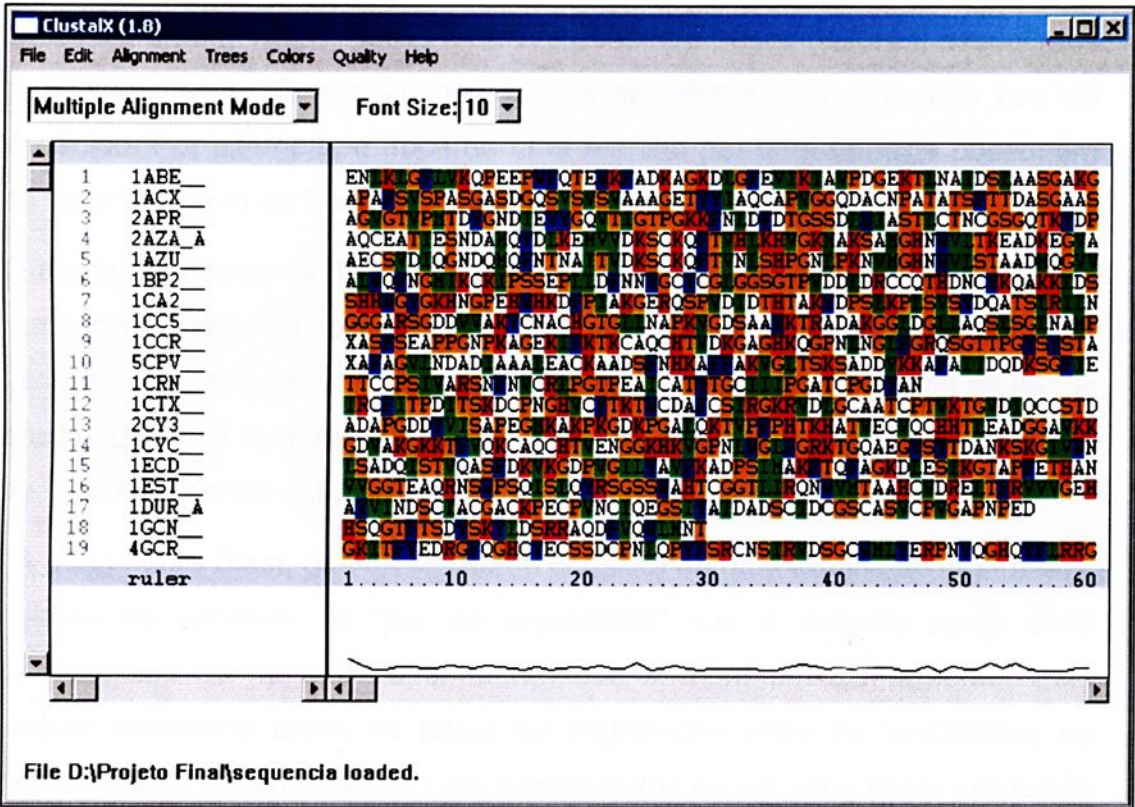


Figura 09 –Exemplo de Alinhamento Realizado pelo CLUSTAL-X. A Sequência de Letras, em cada linha, representa a estrutura primária de cada Proteína.

Para buscar alinhamentos ótimos é necessário definir um critério medido na forma de um escore. O algoritmo ótimo para uma dada definição de escore deverá procurar o alinhamento que maximiza este. Definições distintas de escores resultam em algoritmos diferentes.

Porém, o problema de alinhamento ótimo não se reduz simplesmente à medição do “casamento” entre os aminoácidos das seqüências, é necessário levar em conta o significado biológico. Ou seja, não se pode introduzir gaps de forma aleatória, procurando apenas melhorar o escore. É necessário que as seqüências alinhadas não percam seus significados biológicos.

Esse fato torna o alinhamento um problema mais complexo, pois há a necessidade de distinguir as seqüências com de alto escore e de significância puramente matemática daqueles que possuem significado biológico. Dois

algoritmos bastante conhecidos para realização de alinhamento são o **Fasta** e **Blast** (Rost, 1996).

O *Fasta*, descrito por Lipman e Person, é um método de busca de similaridades locais por meio de emprego de uma matriz de substituição (Lipman & Person, 1985). Essas matrizes possuem pesos (valores) associados a cada substituição de um aminoácido da seqüência alvo por outro tipo de aminoácido. Por exemplo, a troca de uma Alanina por uma Leucina possui um certo valor diferente da troca da Alanina por um Triptofano.

O Fasta concentra-se na busca de casamentos de subsequências curtas de um tamanho específico que podem contribuir para o casamento total. Somente um alinhamento é fornecido para cada uma das seqüências do banco de dados utilizado. Deve-se lembrar que se realiza o alinhamento de uma seqüência teste, que está sendo estudada, com um banco de dados de seqüências.

Já o algoritmo Blast, que na realidade consiste de uma família de algoritmos, baseia-se no conceito de “par de segmentos” que é definido como duas subsequências de mesmo comprimento que formam um alinhamento. Este algoritmo determina todos os pares de segmentos entre as seqüências do banco de dados que apresentam um *escore* acima de um valor limiar pré-fixado pelo usuário. Esses pares de segmentos, chamados de *HSP's (High-Scoring pairs)* são apresentados como saída do programa.

Do ponto de vista operacional, o alinhamento múltiplo é uma generalização “natural” do caso de alinhamento por pares. O problema fundamental do alinhamento múltiplo é a definição precisa da qualidade do alinhamento. Nesse caso, há duas maneiras de definir este *escore*:

- Uma na qual se define o *score* do alinhamento múltiplo como a soma dos *scores* dos alinhamentos de todos os pares não ordenados dentre as seqüências.
- Outra, na qual se define uma seqüência denominada de “consenso” e o *score* do alinhamento múltiplo é definido como a soma dos resultados do alinhamento da seqüência “consenso” com as demais,.



Pode-se definir, de forma simplista, que o objetivo da técnica de alinhamento é otimizar o número de resíduos idênticos nas duas seqüências. O alinhamento da seqüência que queremos estudar, seqüência alvo, contra uma base de dados de seqüências (com estruturas já conhecidas) é um passo inicial e necessário para modelagem por homologia.

2.4.Modelagem por Homologia

A modelagem por homologia, nos últimos 15 anos, tem sido o método para predição de estruturas de proteínas globulares mais utilizado (Barton, 1995) (Lesk & Boswell, 1992) (Rost, 1999, 1998). Essa técnica tem auxiliado a diminuir a diferença entre o número de seqüências conhecidas (estruturas primárias) e de estruturas resolvidas, porém ela não é suficiente.

A principal idéia dessa metodologia baseia-se na observação de que cada seqüência de proteína nativa adota uma única estrutura, porém, muitas seqüências diferentes podem adotar uma mesma estrutura. Ou seja, as proteínas com seqüências similares possuem uma tendência a se enovelarem em estruturas similares. Portanto, se uma seqüência, com estrutura desconhecida (U), possui uma similaridade significativa com uma proteína de estrutura conhecida (T), é possível construir um modelo 3D aproximado para a proteína U, assumindo que esta possui basicamente a mesma estrutura que T.

Há evidências de que a maioria dos pares de proteínas com estruturas similares possuem uma homologia na seqüência menor ou igual a 25%. Essas estruturas homólogas remotas não podem ser identificadas através de técnicas tradicionais de alinhamento (Rost, 1997). Portanto a modelagem por homologia permite a predição de estruturas 3D para apenas 10% a 30% de todas as seqüências de proteínas depositadas no SWISS-PROT (principal banco de dados de seqüência primária).

Consequentemente, outras técnicas computacionais como redes neurais artificiais e algoritmos genéticos vem sendo investigadas (Pendersen & Moulton, 1996) e devem ser exploradas em conjunto com técnicas de homologia e outras técnicas de simulação (não estocásticas) como dinâmica molecular (Kollman, 2001), para predição de estruturas secundárias e 3D, de maneira a aumentar nossa capacidade de predição, seja em número, seja em exatidão (Barton, 1995) (Baldi, 1999).

2.5. Predição 1D (Estruturas Secundárias).

Até o momento, não possuímos habilidade suficiente para prever a estrutura terciária de uma proteína, a partir de sua sequência. Mas podemos prever aspectos mais simplificados da estrutura (Bohr, 1998) (Rost & Sander, 1999). Uma simplificação para o problema é a predição de estruturas de proteínas em uma dimensão (1D), ou seja, a estrutura secundária e/ou a área acessível ao solvente. O nome 1D vem do fato de associarmos a cada aminoácido uma estrutura secundária.

O objetivo usual dos métodos de predição 1D é associar a cada resíduo um padrão estrutural H (α -hélice), E (β -folha) ou L (volta, isto é, uma estrutura não regular), dependente do conjunto de resíduos que lhe são adjacentes, ou seja, de uma janela da sequência. A principal idéia por trás da maioria dos métodos de predição de estrutura secundária é o fato de que segmentos de resíduos consecutivos possuem uma preferência para certos estados de estrutura secundária. Dessa forma, o problema de predição de estrutura torna-se um problema clássico de classificações de padrões tratável por algoritmos de reconhecimento de padrões.

Os algoritmos para predição 1D podem ser, aproximadamente, agrupados da seguinte forma: (i) utilizam informação estatística; (ii) usam propriedades físico-químicas; (iii) fazem uso de padrões de sequência; (iv) utilizam redes neurais e (v) exploram a conservação evolucionária. Hoje, alguns algoritmos utilizam tanto redes neurais como informação evolucionária, ou seja, combinam diferentes recursos (Rost & Sander, 1999). Segundo Rost e outros autores, os



métodos de predição 1D podem ser classificados como de primeira, segunda e terceira geração (Rost & Sander, 1993) (Rost & Sander, 1994). Os métodos de primeira geração eram baseados em propriedades físico-químicas (como o tipo de aminoácido, o peso molecular das proteínas), regras e estatísticas de um único resíduo, ou seja, não utilizavam informações dos resíduos adjacentes

Aos métodos de 2ª geração foram incorporados base de dados maiores e uma estatística baseada em segmentos de resíduos adjacentes, tamanho entre 11 a 21 resíduos (Rost & Sander, 1999). Porém, os métodos de predição de 2ª geração apresentam três problemas principais:

- A exatidão das predições, pelos métodos pertencentes à 2ª geração, é limitada em 60% (porcentagem de resíduos preditos corretamente em um dos três estados: H, E ou L).
- Geralmente, as folhas- β são preditas com menos de 40% de acerto.
- Os segmentos de estruturas secundárias preditos corretamente variam em torno de 50% (do observado).

O primeiro problema possui duas origens: (i) diferentes cristais de uma mesma proteína podem possuir estruturas secundárias diferentes (ii) a formação das estruturas secundárias é determinada, parcialmente, por interações de longo alcance que não são visíveis através de métodos baseados em segmentos de resíduos adjacentes na seqüência. O segundo problema pode ser explicado pelo fato de estruturas do tipo β -folha serem formada mais por contatos não locais do que as estruturas α -hélices, o que dificulta a sua predição.

Alguns dos métodos de 3ª geração procuram tratar esses problemas simultaneamente e são superiores aos métodos de 2ª geração. Os métodos de predição de 3ª geração se diferenciam dos de segunda geração, entre outros fatos por: (a) utilizar informações evolutivas obtidas a partir de alinhamento múltiplo de seqüências; (b) fazer uso de múltiplos níveis de computação (várias redes neurais encadeadas) e (c) pelo treinamento balanceado das redes neurais (Rost, 1999). O exemplo mais conhecido de métodos de 3ª geração é o

PHD, esse método é baseado em redes neurais (isto é, um treinamento com diferentes bases de dados). A maioria dos conceitos utilizados pelos métodos de 3ª geração foi introduzida pelo PHD. Portanto, esse método é um bom exemplo para ilustrar os conceitos dos métodos de 3ª geração.

Rost descreve uma forma de aplicação de redes neurais para a predição 1D das proteínas através do método PHD resumido nas Figura 10 e 11. Para isso é feito um alinhamento múltiplo da seqüência alvo contra uma base de dados utilizando o algoritmo Blast e a seguir realiza-se um refinamento desse alinhamento (na figura esta parte está representada pelas setas entre etapas 1 e 2). Desse extrai-se as informações relevantes (como por exemplo, conservação de resíduos em um dada posição) para a predição (etapa 3). Em seguida, utilizam-se redes neurais que fornecem, como saída a estrutura secundária para cada resíduo da seqüência.

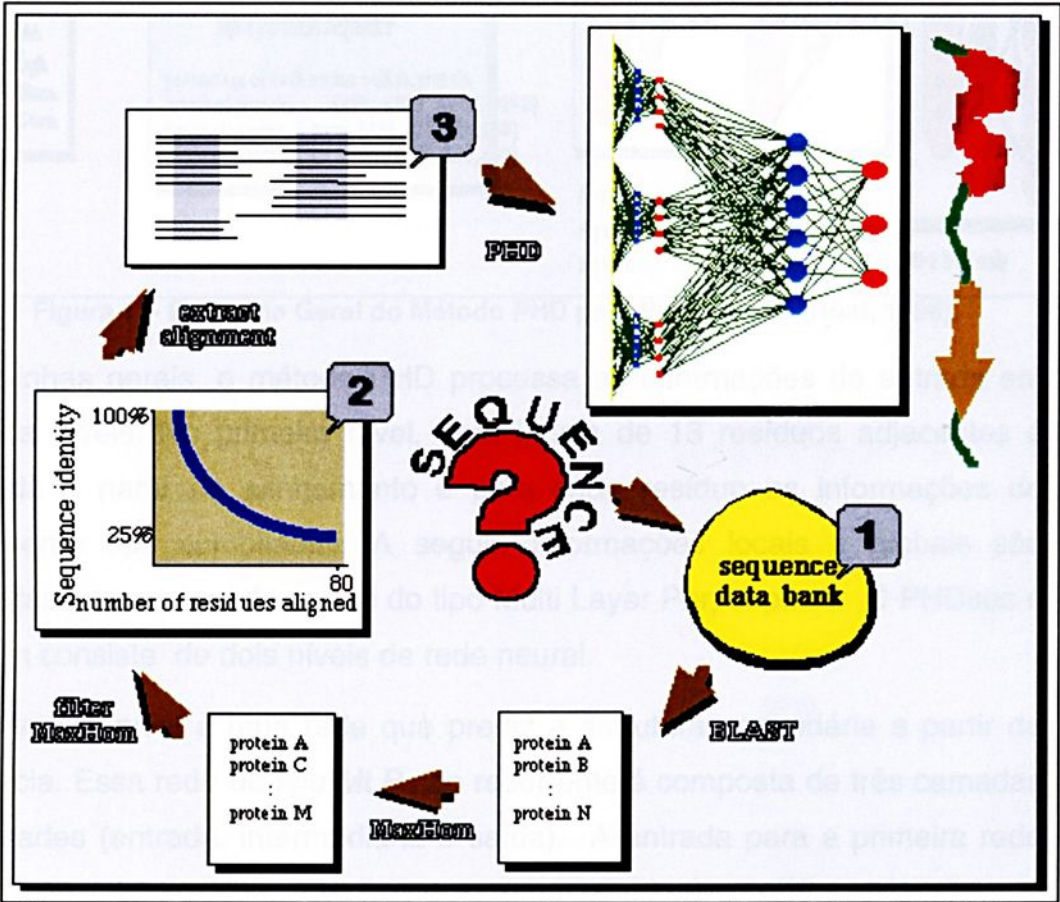


Figura 10 – Esquema Geral do Contexto de uso do PHD (Rost & Sander, 1999).

Existem três tipos de métodos PHD: o PHDsec que realiza predição de estrutura secundária; o PHDacc realiza predição de acessibilidade de solvente e o PHD realiza predição de hélices “transmembrana” (Rost & Sander, 1999).

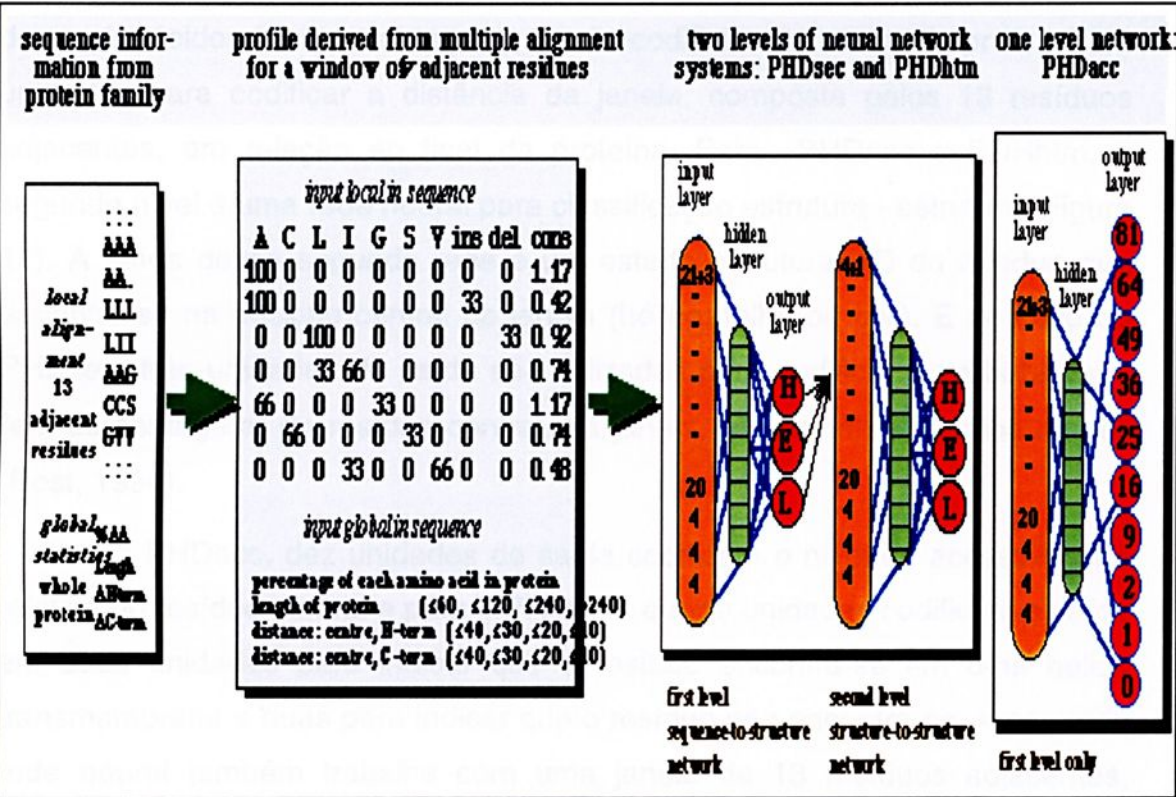


Figura 11- Esquema Geral do Método PHD para Predição 1D (Rost, 1996).

Em linhas gerais, o método PHD processa as informações de entrada em múltiplos níveis. No primeiro nível, uma janela de 13 resíduos adjacentes é escolhida a partir do alinhamento e para cada resíduo as informações do alinhamento são compiladas. A seguir, informações locais e globais são fornecidas para uma rede neural do tipo Multi Layer Perceptron. O PHDsec e PHDhtm consiste de dois níveis de rede neural.

O primeiro nível é uma rede que prediz a estrutura secundária a partir da sequência. Essa rede *do tipo MLP* não recorrente é composta de três camadas de unidades (entrada, intermediária e saída). A entrada para a primeira rede neural consiste de duas contribuições: uma devido à sequência local (uma janela de 13 resíduos adjacentes) e outra devido à sequência global (ver Figura 11)

São utilizadas 24 unidades de entrada para cada uma das 13 posições, 20 descrevem o tipo de aminoácido na posição específica, uma identifica espaços na janela, duas para inserção e remoção naquela posição (informação obtida pelo alinhamento) e uma para representar o peso de conservação. As informações globais são codificadas em 28 unidades, 20 para a composição dos aminoácidos (em porcentagem), 4 para codificar o tamanho da proteína e 4 unidades para codificar a distância da janela, composta pelos 13 resíduos adjacentes, em relação ao final da proteína. Para PHDsec e PDHhtm, o segundo nível é uma rede neural para classificação estrutura - estrutura (Figura 11). A saída dessa segunda rede é um estado estrutural 1D do resíduo que encontra-se na posição central da janela (hélice, folha ou coil). E no caso do PHDsec, três unidades de saída são utilizadas para codificar a probabilidade (em porcentagem) do resíduo central, na janela, ser α -hélice, β -folha ou coil (Rost, 1996).

Para o PHDacc, dez unidades de saída codificam o nível de acessibilidade relativa do resíduo central e para o PHDhtm, quatro unidades codificam a saída em duas unidades para indicar que o resíduo encontra-se em uma hélice transmembrana e duas para indicar que o resíduo não encontra-se. A segunda rede neural também trabalha com uma janela de 13 resíduos adjacentes, realizando a predição do resíduo central. Os dados de saída da primeira rede são reaproveitados como entrada para a segunda rede neural (ou segundo nível) adicionando informações globais (codificadas em 28 unidades, 20 para a composição dos aminoácidos, 4 para codificar o tamanho da proteína e 4 unidades para codificar a distância da janela em relação ao final da proteína) e informações de conservação de peso para cada resíduo da janela. A saída dessa segunda rede neural é igual a da primeira rede neural (Rost, 1996) (Rost & Sander, 1999).

O último nível, chamado de júri, realiza uma média aritmética sobre os resultados (a predição) obtidos por redes distintas e treinadas independentemente. Esse último nível funciona como um filtro e seus dados de entrada são as saídas da segunda rede neural (rede secundária).

Na prática, é como se o júri recebesse a predição de diferentes PHD (redes neurais) e realizasse uma média para decidir qual a estrutura secundária é



associada para o resíduo central na janela da primeira rede. Essas redes distintas possuem a mesma arquitetura (número de neurônios, camadas, codificação e função de ativação); porém são treinadas com conjunto de dados diferentes.

Para acessibilidade de solvente, a codificação da saída não é tão direta. Primeiramente, os valores de acessibilidade são normalizados e depois é projetada sobre dez estados. A saída da primeira rede é idêntica à saída da segunda rede neural que compõem o PHD (Rost & Sander, 1999). As razões para ter sido introduzido um segundo nível foi permitir que a rede levasse informação estrutural de segmentos vizinhos em consideração na predição. O funcionamento do PHD é um bom exemplo da filosofia de funcionamento dos métodos para predição 1D de 3ª geração.

Existem vários servidores para predição 1D como o **Psipred**, **Jpred** e **Jpred2** disponíveis pela Internet (Rost, 1999). O Jpred permite submeter um alinhamento simples ou múltiplo e retorna o resultado da predição obtido utilizando seis algoritmos que exploram informações evolutivas (i.e., informações sobre conservação de aminoácidos retirados do alinhamento). Esse servidor permite dois tipos de entrada: (i) uma família de seqüências de proteínas e (ii) uma única seqüência de proteína. No segundo caso, o JPred cria, de forma automática, um alinhamento múltiplo para depois realizar a predição (Mcguffin & Bryson, 2000) (Cuff et al., 1998). Seis métodos de predição são utilizados, de forma combinada, pelo JPred: *DSC*, *PHD*, *NNSSP*, *PREDATOR*, *ZPRED* e *MULPRED* e as soluções de cada um desses métodos é combinada em um único arquivo (Cuff et al., 1998).



2.6. Predição 2D e 3D (Estruturas Terciárias).

Fornecidos todos os contatos ou distâncias, a estrutura terciária de uma proteína poderia ser reconstruída por dinâmica molecular. Isso é utilizado para determinação de estruturas 3D através de espectroscopia de RNM (Ressonância Magnética Nuclear) que fornece dados experimentais das distâncias entre os prótons (H).

Obviamente, uma predição com sucesso da estrutura secundária (1D) é um pré-requisito para uma predição bem sucedida de uma parte de todos os contatos inter-resíduos (predição 2D) (Rost, 1999). Contudo, contatos que foram preditos a partir de associação de estruturas secundárias entre resíduos próximos na seqüência e é necessário predizer também os contatos entre resíduos distantes na seqüência.

Uma simplificação do problema de predição de contatos inter-resíduos é focalizar-se na predição de contatos entre resíduos em um segmento. Esse esforço é motivado pelo fato de que se espera que a interação em uma estrutura secundária (*motif*) seja mais específica do que os contatos entre resíduos distante na cadeia (longo alcance) e, portanto mais fácil de predizer. O único método publicado para predição de contatos interestruturas é baseado em potenciais de força média similar aos utilizados para avaliar a tendência de formar estrutura (Rost, 1999). Ao utilizar informações evolucionárias, ou seja, informações obtidas por alinhamento múltiplo, os contatos inter-resíduos podem ser preditos com uma taxa de acerto maior (Rost, 1998).

Um avanço teórico importante na predição 3D, nos anos recentes, foi o desenvolvimento de campos de força média. Antes desses potenciais, a predição de estrutura era normalmente realizada com potenciais “físicos”, i.e., um potencial torsional, que representa a liberdade de rotações da molécula em torno de cada ligação química, e de um potencial dito “non-bounded”, entre átomos ou grupos ligados quimicamente, constituído de um termo do tipo Van der Waals e do termo eletrostático. Os potenciais “físicos” descrevem a energia interna da molécula e têm sido utilizados com sucesso para refinar estruturas determinadas experimentalmente. Ao contrário destes, os potenciais de força



média extraídos de base de dados de proteínas descrevem a energia livre da molécula. Os potenciais “físicos” tem sido utilizados com muito sucesso para refinar estruturas determinadas experimentalmente.

Algoritmos genéticos também vêm sendo utilizados com sucesso na predição 3D de pequenas proteínas e peptídeos a partir de sua sequência primária, minimizando a função de energia potencial da macromolécula. Vários autores têm obtido resultados consideráveis, no esforço de predizer a estrutura terciária de trechos de proteínas utilizando algoritmo genético (Sun, 1995) (Dandekar, 1998).

2.7.Métodos Computacionais e o Problema de Predição.

Existem diversas técnicas de simulação computacional que podem e vêm sendo empregadas para estudar estruturas de proteínas como: Monte Carlo, Recozimento Simulado (*Simulated Annealing*), Algoritmos Genéticos e Redes Neurais Artificiais, Machine Learning, Teoria dos Grafos e Dinâmica Molecular (Compiani et al., 1998) (Cui, 1998) (Keneller, 1990) (Kono & Doi, 1994) (Le Grand & Merz, 1994) (Le Grand & Merz, 1993) (Maclin & Shalvick, 1993) (Murray, 1994). Algumas dessas técnicas são estocásticas e outras são determinísticas.

Cada técnica possui suas vantagens e desvantagens e pode ser utilizada em conjunto com outra para o problema de estudo conformacional de proteínas. A seguir, discutiremos brevemente a aplicação de técnicas “aprendizado de máquina”.

2.7.1. Machine Learning (Aprendizado de Máquina).

Essa técnica é baseada em regras, descritas através de uma lógica de primeira ordem, que são deduzidas a partir de relações conhecidas entre elementos de informação que são subsequenteiramente generalizadas. As regras são o componente central dessa abordagem.

Essas regras são depositadas em uma base de dados utilizando a linguagem de programação Prolog. Essa base, por sua vez, faz parte dos sistemas especialistas que são projetados com o objetivo de realizar a predição e avaliar as regras entre as estruturas primárias e secundárias de proteínas, isto é, aprimorar a qualidade de predição de estruturas secundárias (Bohm, 1996).

O PROMIS (PROtein Machine Induction System) é um programa baseado na abordagem de aprendizado de máquina (i.e., é um sistema especialista); com ele se obteve 60% na predição de três estados (hélice, folha e coil) na estrutura secundária para proteínas de domínio desconhecido. Um método similar foi desenvolvido para a dedução de propriedades de aminoácidos em uma sequência padrão pré-definida com uma estrutura terciária conhecida (Bohm, 1996).

Rost apresenta um estudo comparando a qualidade da técnica “machine learning” com métodos tradicionais e redes neurais (Rost, 1998). O programa resultante GOLEM utiliza lógica de programação indutiva e tem apresentado bons resultados na predição de elementos de estrutura secundária a partir da sequência primária. Sistemas baseados na teoria aprendizado de máquina também foram desenvolvidos para prever propriedades funcionais de enzimas o relacionamento estrutura-atividade de ligantes em proteínas.

Vários autores concluem que as regras deduzidas a partir de algoritmos de aprendizado de máquina, juntamente com intervenção de especialistas humanos, é uma poderosa ferramenta, especialmente, na investigação de propriedades estereoquímicas de macromoléculas biológicas (Rost, 1999). Essa técnica foi bastante explorada e utilizada durante algum tempo,

influenciando a área de predição de estruturas e, portanto possui uma importância histórica.

Capítulo 3

Redes Neurais Artificiais (RNA)

2.1. Considerações Iniciais

O estudo das redes neurais artificiais é um campo extremamente interdisciplinar tanto em relação ao seu desenvolvimento como em relação à sua aplicação. Alguns exemplos de áreas onde essas redes têm sido aplicadas são: reconhecimento de padrões, controle, reconhecimento de voz, medicina, reconhecimento de faces e predição de estrutura de proteínas. As RNAs são modelos de computação com características particulares como: capacidade de aprender, generalizar, adaptar e organizar dados. São capazes de armazenar informações e de recuperar essas informações quando necessário (Haykin, 1999).

Este Capítulo tem por objetivo discutir os conceitos básicos de redes neurais, os tipos de Redes Neurais Artificiais (RNA) e as formas de treinamento. Na Seção 2.2 introduzimos os conceitos básicos de teoria das redes neurais. Na Seção 2.3 são discutidos três tipos de RNA (MLP, Hopfield e Redes sem Peso), e nas duas últimas seções são exibidas

Capítulo 3

Redes Neurais Artificiais (RNA)

3.1.Considerações Iniciais

O estudo de redes neurais artificiais é um campo extremamente interdisciplinar tanto em relação ao seu desenvolvimento como em relação a sua aplicação. Alguns exemplos de áreas aonde redes neurais vem sendo aplicadas são: processamento de sinais; controle; reconhecimento de padrões; medicina; reconhecimento de fala; e predição de estrutura de proteínas. As RNAs são modelos de computação com características particulares como: *capacidade de aprender, generalizar, agrupar e organizar dados; não linearidade; mapeamento entre entrada e saída, adaptabilidade, informação contextual, tolerância a falhas* (Haykin, 1999).

Esse Capítulo têm por objetivo discutir os conceitos básicos de redes neurais, os tipos de Redes Neurais Artificiais (RNA) e as formas de treinamento/apredizagem. Na Seção 3.2 introduzem-se os conceitos básicos da teoria de redes neurais. Na Seção 3.3 são discutidos três tipos de RNA (MLP, Hopfield e Redes sem Peso), essas duas primeiras são exploradas

nesse trabalho de doutorado. Na Seção 3.4 apresentam-se alguns conceitos básicos sobre o processo de aprendizagem/treinamento das redes.

3.2.Conceitos Básicos

O sistema nervoso é formado por um conjunto extremamente complexo de células simples chamadas neurônios. Essas células possuem um papel essencial na determinação do funcionamento do corpo humano e do raciocínio (Braga et al., 1998) (Fausset, 1994). Um neurônio típico é formado por *dendritos* (conjunto de terminais de entrada), pelo corpo central (*soma*) e pelo *axônio* (um longo terminal de saída). A Figura 12 ilustra os componentes de um neurônio.

Os neurônios comunicam-se através de sinapses que são as regiões onde dois neurônios entram em contato e através da qual os impulsos nervosos são transmitidos entre eles. Os impulsos recebidos por um neurônio A, em um determinado momento, são processados, e atingindo um limiar específico de ação, o neurônio A dispara um sinal que flui do corpo celular para o axônio, o qual pode ser conectado a um dendrito de um outro neurônio B.

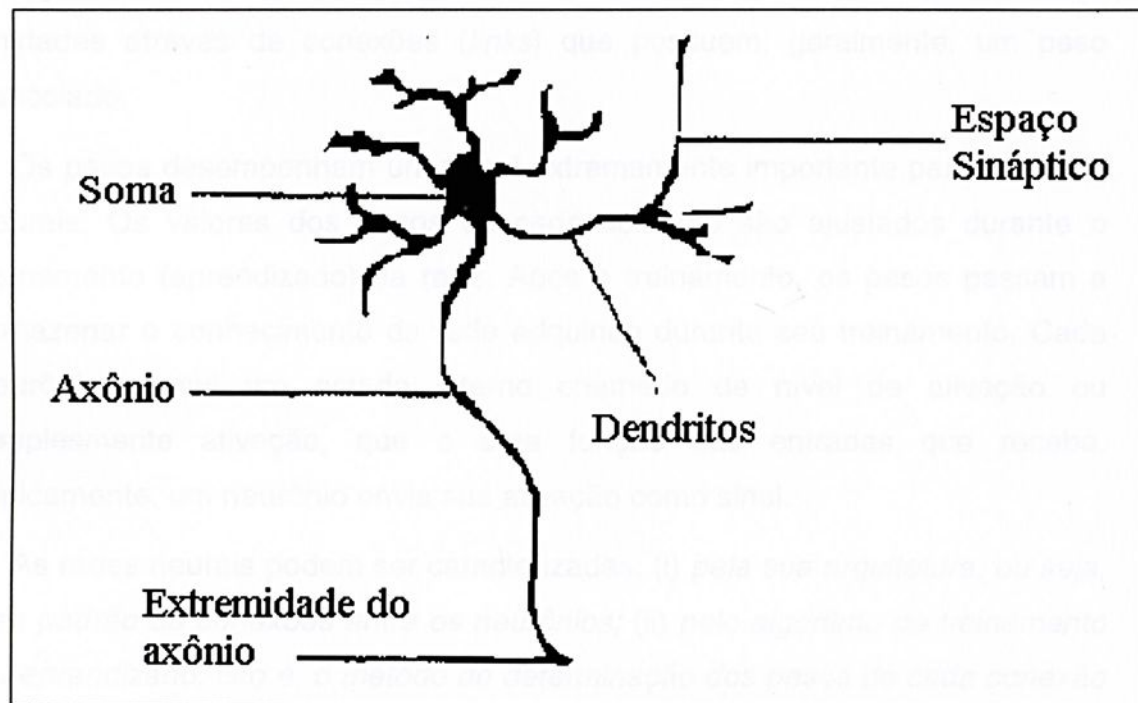


Figura 12 – Esquema de um neurônio Biológico.

O neurotransmissor pode diminuir ou aumentar a polaridade da membrana pós-sináptica, inibindo ou excitando a geração dos pulsos no neurônio B. Este

processo depende de vários fatores, tais como a geometria da sinapse e o tipo de neurotransmissor (Braga et al. 1998) (Haykin, 1999) (Goonatilake & Khebbal, 1995). Em média, cada neurônio forma entre mil e dez mil sinapses. O cérebro humano possui cerca de 10^{11} neurônios, e o número de sinapses é maior que 10^{14} , possibilitando a formação de redes muito complexas.

Redes Neurais Artificiais são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento ("aprendem") através da experiência. Uma grande rede neural artificial pode possuir centenas ou milhares de unidades de processamento, já o cérebro humano pode possuir centenas de bilhões de neurônios (Braga, 1998) (Fausset, 1994). Os modelos de neurônios ou nós das redes neurais artificiais são baseados nos neurônios biológicos.

Pode-se definir uma rede neural artificial como um sistema de processamento de informações que possui certas características de desempenho similares com as redes neurais biológicas. Portanto, uma rede neural consiste de um número grande de elementos de processamento simples, chamado neurônio, célula ou nó. Cada neurônio é conectado a outras unidades através de conexões (*links*) que possuem, geralmente, um peso associado.

Os pesos desempenham um papel extremamente importante para as redes neurais. Os valores dos pesos de cada conexão são ajustados durante o treinamento (aprendizado) da rede. Após o treinamento, os pesos passam a armazenar o conhecimento da rede adquirido durante seu treinamento. Cada neurônio possui um estado interno chamado de nível de ativação ou simplesmente ativação, que é uma função das entradas que recebe. Tipicamente, um neurônio envia sua ativação como sinal.

As redes neurais podem ser caracterizadas: (i) *pela sua arquitetura, ou seja, seu padrão de conexões entre os neurônios*; (ii) *pelo algoritmo de treinamento ou aprendizado, isto é, o método de determinação dos pesos de cada conexão* e (iii) *e pela função de ativação* (Kòvacs, 1996).



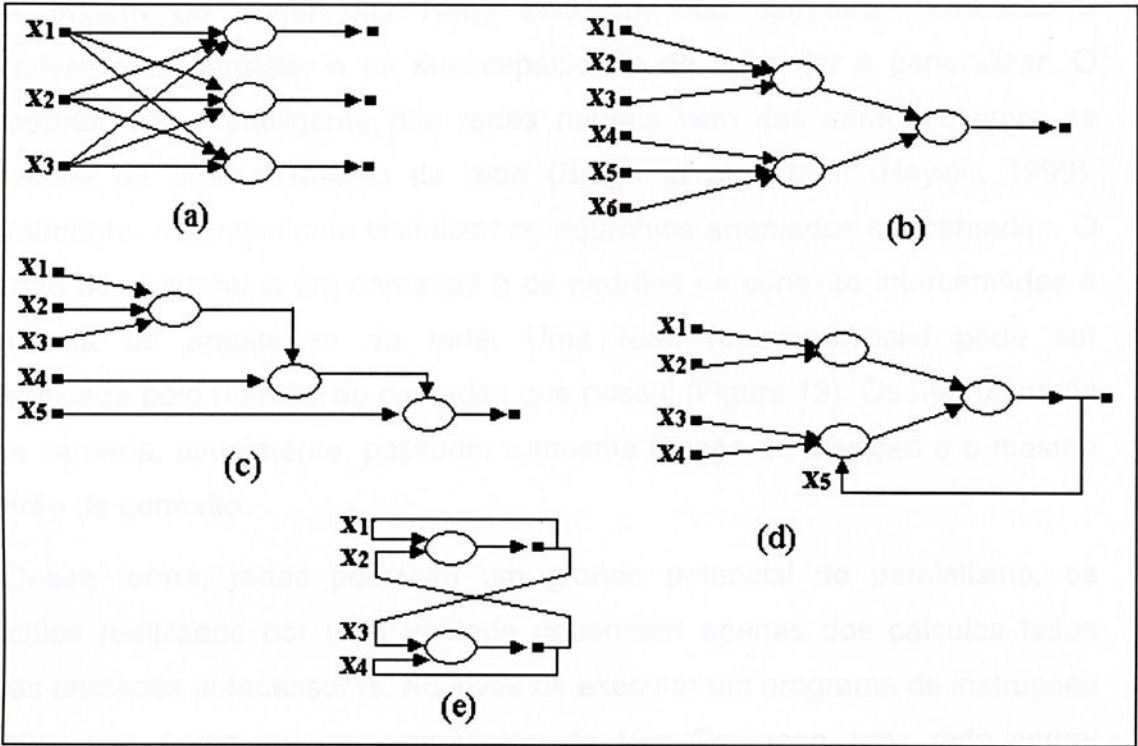


Figura 13 – Exemplos de uma arquiteturas recorrentes (d e e) e não recorrentes (a,b e c)

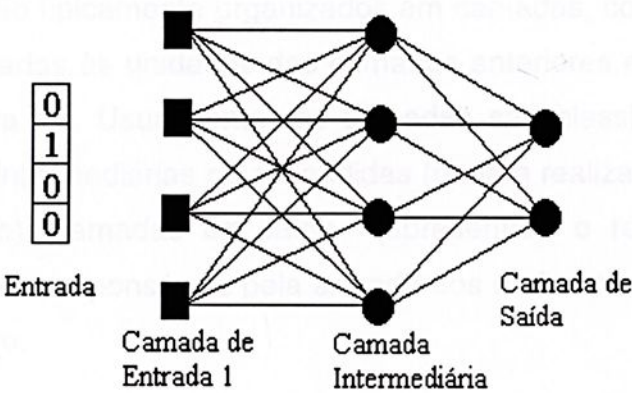


Figura 14- Exemplo de uma rede MLP completamente conectada, não recorrente e com uma camada intermediária de neurônios.

A origem de poder das Rnas está em sua estrutura distribuída e massivamente paralela; e na sua capacidade de aprender e generalizar. O comportamento inteligente das redes neurais vem das iterações entre as unidades de processamento da rede (Braga et al., 1998) (Haykin, 1999). Geralmente, é conveniente visualizar os neurônios arranjados em camadas. O arranjo dos neurônios em camadas e os padrões de conexão intercamadas é chamada de arquitetura da rede. Uma rede neural artificial pode ser classificada pelo número de camadas que possui (Figura 13). Os neurônios de uma camada, usualmente, possuem a mesma função de ativação e o mesmo padrão de conexão.

Dessa forma, redes possuem um grande potencial de paralelismo, os cálculos realizados por uma unidade dependem apenas dos cálculos feitos pelas unidades antecessoras. Ao invés de executar um programa de instruções seqüenciais, como em um computador de *Von Neumann*, uma rede neural responde às entradas que lhe são apresentadas em paralelo (Braga, 1998). Os resultados não são armazenados em uma determinada posição de memória, mas são dados pelo estado geral da rede depois que ela atinge a condição de equilíbrio, a convergência. Assim, o conhecimento está na forma como as unidades estão interconectadas e na importância (peso) de cada entrada em uma determinada unidade.

Os neurônios são tipicamente organizados em camadas, com unidades que podem ser conectadas às unidades das camadas anteriores e posterior como ilustrado na Figura 14. Usualmente, as camadas são classificadas em três grupos: camadas intermediárias ou escondidas (onde é realizada a maior parte do processamento), camadas de saída (apresentam o resultado final) e camadas de entrada responsáveis pela entrada dos dados para a rede realizar seu processamento.

A forma como as unidades de processamento encontram-se conectadas também faz parte da arquitetura da rede. Os neurônios podem estar conectados intracamada e intercamadas. Na Figura 13d, apresenta-se uma rede neural organizada em uma única camada, onde os neurônios possuem laços (conexões recorrentes). Deve-se lembrar que a cada conexão entre dois neurônios é associado um peso que armazena o conhecimento da rede

adquirido durante seu treinamento. O peso realiza o papel da *sinapse*, aumentando ou diminuindo o valor do sinal de cada conexão.

Na Figura 14, está representada uma rede neural completamente conectada organizada em três camadas: uma camada de entrada de dados, uma camada intermediária (classificação) e uma camada de saída. Em uma rede neural totalmente conectada cada neurônio da camada $i-1$ está ligado a todos os neurônios da camada i . Pode-se observar que a rede ilustrada na Figura 14 não possui laços, ou seja, uma conexão entre um neurônio que esteja em uma camada i com neurônios que se encontram em camadas anteriores $j < i$. Esse tipo de rede é chamado de não recorrente ou "*feed forward*". A presença ou não de laços (*loops*) em uma RNA é uma característica importante para sua arquitetura.

As operações básicas de uma RNA envolvem somar os sinais que entram multiplicados pelos pesos e aplicar uma função de ativação para determinar a saída da rede. Para a camada de entrada de dados, a função de ativação, geralmente, é a função identidade. Alguns tipos comuns de função de ativação são: função identidade, função escada, função sigmoidal binária e função sigmoidal polar.

A maioria dos modelos de redes neurais possui alguma regra de treinamento responsável por ajustar os pesos de suas conexões de acordo com os padrões apresentados para a rede, essas regras de treinamento serão discutidas na Seção 3.4. Existem vários tipos de redes neurais artificiais, cada um com sua própria estrutura e com seu algoritmo de aprendizado, como: Perceptron, Multi Layers Perceptron Multi-Camadas, Radial Basis Function (RBF), Adaline, Memórias Matriciais (modelo de Hopfield) e Redes neurais sem peso.

Vários trabalhos, aplicando redes neurais, para estudar o problema de predição de estrutura são descritos na literatura (Barton, 1995) (Baucon et al., 1996) (Bohr et al., 1998) (Cai et al., 2000) (Cai et al., 1999) (Chandonia & Karplus, 1996) (Chandonia & Karplus, 1999) (Compiani, 1998) (Cuff & Barton, 2000) (Cuff & Barton, 1999) (Cui et al., 1998). Em alguns deles, redes do tipo MLP (Multi Layer Perceptron) foram aplicadas para predição de estruturas secundárias como hélice- α folha- β . Outros trabalhos utilizam RNA do tipo



Hopfield para prever a sequência de proteínas e a conformação das cadeias laterais dos resíduos que compõem o core da proteína.

3.3.Tipos de Redes Neurais Artificiais (Exemplos de RNAs)

3.3.1. Perceptron

O neurônio de MacCulloch-Pitts é talvez o neurônio artificial proposto mais antigo, ele apresenta características importantes encontradas em muitas redes neurais (Fausset, 1994). Os atributos desse neurônio podem ser resumidos em:

- A ativação do neurônio é binária. A qualquer passo do tempo, o neurônio dispara (possui ativação igual a um) ou não dispara (ativação igual a zero).
- Os neurônios são conectados por caminhos diretos com peso.
- Uma conexão é excitatória se o peso é positivo, caso contrário é inibitório. Todas as conexões em um neurônio particular possuem o mesmo peso.
- Cada neurônio possui um limiar de ativação ("threshold") fixo, de forma que se a entradas da rede para o neurônio for maior que o limiar, esse se torna ativo.
- Leva algum tempo para o sinal passar de uma conexão para outra.

Redes do tipo Perceptron tiveram um grande impacto nos primeiros anos das redes neurais artificiais. Existe um número de diferentes perceptrons (Fausset, 1994). Tipicamente, os perceptrons originais tinha três camadas de neurônios: unidades de sensores, unidades associadoras e unidades de resposta, formando um modelo próximo ao da retina. A Figura 16 ilustra a arquitetura de um Perceptron simples para classificação de padrão. Os círculos representam as unidades de processamento (neurônios) e as linhas as conexões entre os neurônios.

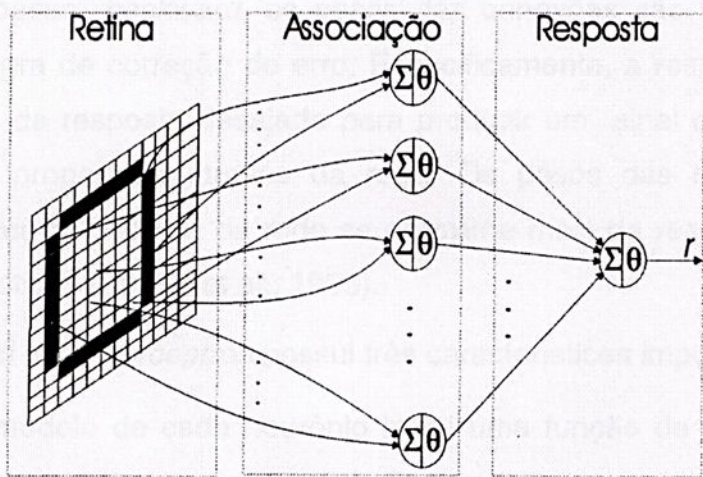


Figura 15- Ilustração de uma Rede Perceptron para Classificação de Padrões. Os círculos representam os neurônios (unidades de processamento).

3.3.2.Redes Multi Layer Perceptron (MLP)

As MLP são uma classe importante de redes neurais e vêm sendo empregadas com muito sucesso para resolver problemas diversos e difíceis, através de treinamento supervisionado e utilizando um algoritmo bastante popular conhecido como retropropagação (*back-propagation*) (Braga et al., 1998).

Tipicamente, essas redes consistem de uma camada de entrada de dados, uma ou mais camadas intermediárias (ou, escondidas) de neurônios e uma camada de saída. O sinal de entrada da rede propaga-se em uma direção *forward*. Basicamente, o algoritmo de treinamento dessa rede (*back-propagation*) consiste de dois passos básicos: um passo *forward* e um passo *backward*. No primeiro, um padrão ativo (vetor de entrada) é aplicado aos neurônios da camada de entrada e seu efeito propaga-se pela rede através das camadas até que um resultado seja produzido pela camada de saída. As camadas intermediárias desse tipo de rede são responsáveis pelo processamento principal.

No segundo passo, *backward*, os pesos das conexões são ajustados de acordo com a regra de correção de erro. Especificamente, a resposta real da rede é subtraída da resposta desejada para produzir um sinal de erro. Esse sinal de erro é propagado através da rede. Os pesos das sinapses são ajustados para que a resposta da rede se aproxime mais da resposta correta em termos estatísticos (Braga et al., 1998).

Uma rede *multi layer perceptron* possui três características importantes:

- O modelo de cada neurônio inclui uma função de ativação não linear.
- A rede possui uma ou mais camadas intermediárias que não são partes da entrada ou saída da rede.
- As redes apresentam um alto grau de conectividade.

Essa classe de redes é utilizada com bastante sucesso para problemas de reconhecimento e classificação de padrões e, portanto, são indicadas para problemas de predição de estrutura secundária que são essencialmente classificação de padrões.

A Figura 14 ilustra um exemplo de uma rede do tipo MLP com três camadas (uma camada de entrada, uma camada intermediárias e uma camada de saída), onde novamente os círculos representam os neurônios (unidades de processamento) e as linhas representam as conexões entre os neurônios. Detalhes sobre o algoritmo *retropropagação* ou sobre o formalismo das redes MLP podem ser encontrados em (Haykin, 1999). Uma rede Multi Layer Perceptron (MLP) pode ser recorrente ou não e completa ou parcialmente conectada.

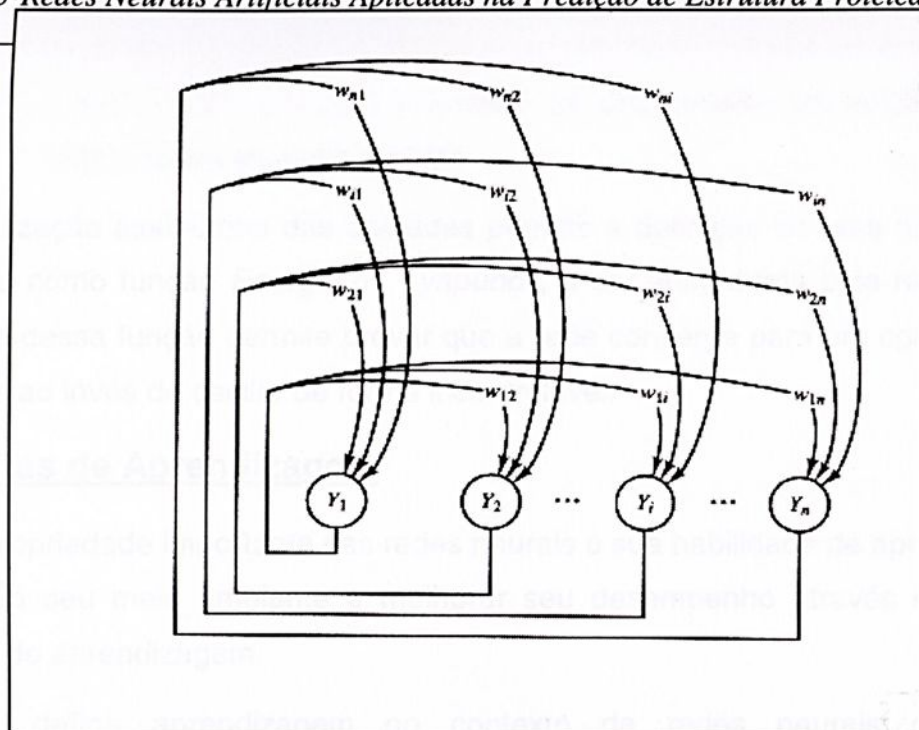


Figura 16- Ilustração de uma Rede Hopfield Discreta.

3.3.3.Redes Hopfield

As redes de Hopfield (ou modelo de Hopfield) consistem de um conjunto de neurônios e um conjunto de unidades de atraso, formando um sistema de laços (*loop*) múltiplos, como ilustrado na Figura 16 (Braga et al., 1998). O número de *laços feedback* é igual ao número de neurônios. Basicamente, a saída de cada neurônio é *feedback* para os demais na rede. Na prática, cada um dos n neurônios possui $n-1$ entradas *feedback*.

As redes de Hopfield podem operar de forma discreta ou contínua, dependendo do modelo adotado para descrever os neurônios. O modelo discreto é baseado no modelo de MacCulloch-Pitts e o discutiremos brevemente. Essa rede é completamente conectada no sentido que cada unidade está conectada a todas as outras unidades. Um rede de Hopfield discreta tem uma matriz de peso simétrica e com a diagonal igual a zero, i.e., $w_{ij} = w_{ji}$ e $w_{ii} = 0$. As redes de Hopfield são um tipo especial de redes autoassociativas e nessas redes:

- Apenas uma unidade atualiza sua função de ativação em um passo de tempo (baseado nos sinais que recebe das outras unidades).

- Cada unidade continua a receber um sinal externo em adição aos outros sinais de cada unidade.

A atualização assíncrona das unidades permite a definição de uma função, conhecida como função *Energia* ou *Lyapunov*, a ser encontrada pela rede. A existência dessa função permite provar que a rede converge para um conjunto de estado ao invés de oscilar de forma incontrolável.

3.4.Regras de Aprendizagem

Uma propriedade importante das redes neurais é sua habilidade de aprender a partir do seu meio ambiente e melhorar seu desempenho através desse processo de aprendizagem.

Haykin define aprendizagem no contexto de redes neurais como: “Aprendizado é um processo pelo qual os parâmetros livres da rede são ajustados/adaptados através de um processo de estimulação pelo ambiente no qual a RNA está embutida. O tipo de aprendizado é determinado pelas maneiras com que os parâmetros são ajustados” (Haykin, 1999). Essa definição do processo de aprendizagem implica na seguinte seqüência de eventos:

- *A Rede Neural é estimulada pelo ambiente.*
- *A Rede modifica seus parâmetros livres como resultado do estímulo.*
- *A RNA responde de uma forma nova ao ambiente, devido às mudanças que tenham ocorrido em sua estrutura interna (ou seja, em seus parâmetros livres).*

O processo de aprendizado varia de acordo com a topologia da rede e ajusta, iterativamente, os pesos da rede. Denomina-se algoritmo de aprendizado, a um conjunto de regras bem definidas para a solução do problema de aprendizado de uma rede neural. Existem diversos algoritmos de aprendizado como: correção de erro, Hebbiano, Competitivo e Boltzmann. Os algoritmos diferem entre si, principalmente, pela forma com a qual ajustam (atualizam) os pesos da rede(Braga et al., 1998) (Haykin, 1999)..

Outro fator importante no aprendizado de uma rede neural é a forma como ela se relaciona com o ambiente. Nesse contexto existem os seguintes *paradigmas de aprendizado*: Aprendizado Supervisionado (a resposta para cada entrada deve ser conhecida); Aprendizado não supervisionado (não se conhece a resposta para cada entrada) e Aprendizado por reforço. Um paradigma de aprendizado pode possuir vários algoritmos de aprendizado (Fausset, 1994). A título de ilustração, descreveremos o algoritmo de aprendizado por correção de erro que pertence ao paradigma de aprendizagem supervisionada. Esse algoritmo de aprendizado é bastante utilizado pelas redes neurais e pertence ao paradigma supervisionado.

3.4.1. Aprendizado por Correção de Erro

Para ilustrar essa regra/algoritmo de aprendizado, considere o caso simples de um neurônio k constituindo o único nó computacional na camada de saída de um rede neural *feedforward*. O Neurônio k recebe um conjunto de sinais (um vetor) $X(n)$ produzido por uma ou mais camadas intermediárias, que, por sua vez, recebem um vetor de sinais (vetor de entrada) da camada de entrada de dados. O índice n denota o passo de tempo no processo iterativo de ajustar os pesos da *sinapse* do neurônio k . O sinal de saída do neurônio k é representado por $y_k(n)$. Essa saída é comparada com a resposta desejada denotada por $d_k(n)$. E um sinal de erro, $e_k(n)$ é definido como:

$$(i) \quad e_k(n) = d_k(n) - y_k(n).$$

Utilizando o sinal de erro, a rede ajusta os pesos das conexões com o neurônio k , da forma que a saída fique mais próxima do valor de saída desejado. Esse objetivo é alcançado, minimizando uma função custo. Esse processo ou algoritmo de aprendizado é conhecido como: *aprendizagem por correção de erro*. Em particular, a minimização da função custo $E(n)$ leva a uma regra de aprendizagem conhecida como regra delta ou *Windrow-Hoff*. Seja $w_{kj}(n)$ o valor do peso da sinapse entre do neurônio k e o neurônio j no passo de tempo n . De acordo com essa regra, o ajuste do peso da sinapse no passo n será:

$\Delta w_{kj}(n) = \eta e_k(n) x_k(n)$, onde η é uma constante positiva, chamada de taxa de aprendizagem e $x_k(n)$ é o valor de entrada do neurônio. Tendo computado

Algoritmos Genéticos

4.1.Considerações Iniciais.

A otimização utilizando algoritmos genéticos pode ser aplicada a diversos tipos de problemas, desde ajuste de curvas (parametrização), passando pelo problema do cacheiro viajante à predição da estrutura terciária de uma proteína. Algoritmos Genéticos (AGs) oferecem uma forma eficaz de busca conformacional. Essa técnica foi introduzida, inicialmente, por Holland para simular o processo de seleção natural. Porém, foi aceita rapidamente como um método geral de otimização e vêm sendo aplicado na predição de estruturas moleculares (principalmente proteínas e peptídeos) e na otimização de conformações de macromoléculas baseados na função de energia potencial (Goldberg, 1989).

Pesquisadores vêm desenvolvendo e aprimorando, na última década, vários algoritmos genéticos para busca conformacional de proteínas (Desjarlais & Handel, 1999) (Dandekar & Argos, 1998) (Dandekar & Argos, 1992) (Cui et. al., 1998) (Friedrichs & Wolynes, 1996) (Le Grand & Merz, 1993) (Le Grand & Merz, 1994) (Penderson & Moulton, 1997) .

Esse trabalho de doutorado explorou o uso de algoritmos genéticos e de bibliotecas de rotâmeros na predição de estruturas protéicas. Dessa forma, esse Capítulo discute, em um primeiro momento, os conceitos básicos de algoritmos genéticos e sua filosofia de funcionamento, e a seguir, apresenta aplicações dessa técnica na predição de estruturas protéicas, descritas na literatura.

A Seção 4.2 apresenta conceitos básicos da teoria de algoritmos genético. Na Seção 4.3 é ilustrado um exemplo de aplicação de algoritmo genético e de seu funcionamento.

4.2. Conceitos Básicos

Algoritmos Genéticos (AGs) são algoritmos de otimização e busca, baseados nos mecanismos de seleção natural proposta por Charles Darwin (Goldberg, 1989). Enquanto os métodos de otimização e busca convencionais trabalham geralmente de forma seqüencial, avaliando a cada instante uma possível solução, os AGs trabalham com um conjunto de possíveis soluções simultaneamente. De acordo com Goldberg, os algoritmos genéticos diferem dos métodos tradicionais de busca e otimização em três aspectos principais:

- Trabalham com uma codificação do conjunto de parâmetros e não com os próprios parâmetros.
- Trabalham com uma população de soluções candidatas simultaneamente e não com um única solução.
- Utilizam regras de transição probabilísticas, e não determinísticas.

Na terminologia desses algoritmos, uma solução candidata é chamada de indivíduo e ao conjunto de indivíduos simultaneamente avaliados é dado o nome de população. A cada indivíduo é associado um grau de adaptação, que mede a capacidade de solução, representada pelo indivíduo para resolver um dado problema. Os algoritmos genéticos não garantem a solução ótima, mas geralmente encontram “soluções quase ótimas” em tempos factíveis (Braga et al., 1998).

Esta técnica emprega uma estratégia de busca paralela e estruturada, com um componente aleatório forte, que é voltada em direção à busca por pontos de “maior aptidão”, ou seja, pontos nos quais a função a ser minimizada (ou maximizada) possui valores baixos (ou altos). Apesar de aleatórios, eles não são caminhadas aleatórias não direcionadas, pois utilizam informações históricas para definir novos pontos, onde são esperados melhores

desempenhos (valores). Isto é realizado através de processos iterativos, onde cada iteração é chamada de geração. Durante cada iteração (geração), princípios de seleção e reprodução são aplicados às populações candidatas à solução (BRAGA et al., 1998).

O primeiro aspecto (passo) a ser considerado para a utilização de algoritmos genéticos como ferramentas para solução de problemas é a representação destes em uma estrutura na qual os algoritmos possam trabalhar adequadamente. Dá-se o nome de genótipo ou cromossomo ao indivíduo codificado de acordo com essa estrutura. Tradicionalmente, os indivíduos são representados por vetores de números binários (BRAGA et al., 1998). O processo de seleção determina quais indivíduos da população podem participar da fase de reprodução. Esses indivíduos são selecionados de acordo com uma probabilidade dada pelos seus índices ou notas de aptidão. Assim indivíduos com maior aptidão possuem uma grande chance de participar da reprodução (Goldberg, 1989). Um método de seleção muito utilizado é o chamado Método da Roleta, onde cada indivíduo ocupa, em uma roleta, uma área proporcional ao seu índice de aptidão.

Durante a seleção, a roleta é girada N vezes, selecionando N indivíduos para participarem da fase de reprodução. Os indivíduos escolhidos participam da fase de reprodução, onde podem ser combinados ou modificados, produzindo os indivíduos da nova geração e os principais operadores de reprodução utilizados são: *cruzamento (crossover)* e *mutação*. Eles são utilizados para assegurar que a nova geração apresente novas soluções, mantendo características desejáveis adquiridas em gerações anteriores.

Ao utilizar os algoritmos genéticos, existem parâmetros que devem ser observados, pois podem influenciar no comportamento. Entre eles encontra-se: o tamanho da população, a taxa de *cruzamento* e a taxa de *mutação*.

4.3.Exemplo de Aplicação

Um algoritmo genético começa pela codificação de k variáveis independentes de um problema de otimização como genes em um cromossomo. Por exemplo, na busca conformacional, os genes poderiam ser a conformação determinando os ângulos diedrais de uma molécula, como



ilustrado na Figura 17. A seguir, uma população inicial de N cromossomos [chamada de P(t)] é criada com valores randômicos para cada gene de cada cromossomo. Depois da inicialização, uma função calcula a energia de cada conformação e utiliza-a como critério de seleção dos cromossomos para passarem ou não para a próxima geração.

A seguir, a cada passo, um quantidade M de cromossomos da população P(t) é selecionada e recombinada através do operador de *cruzamento*.

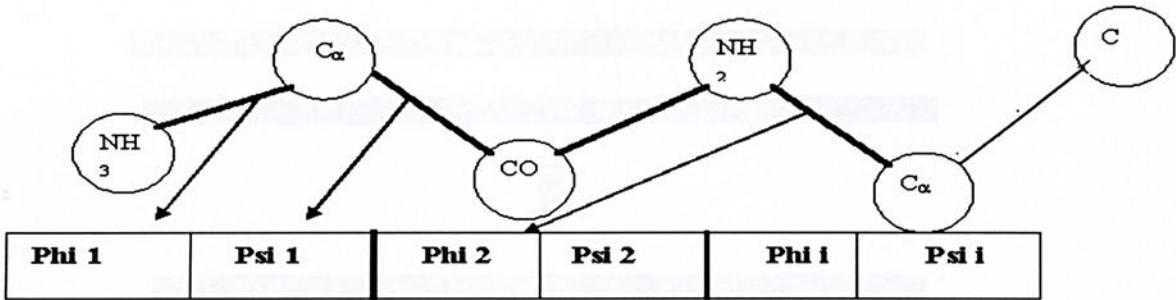


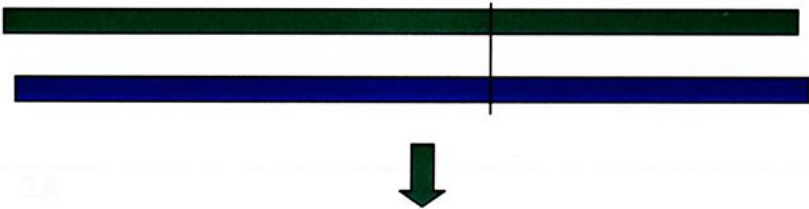
Figura 17 – Exemplo de codificação de uma conformação de proteína em um gene (cromossomo) de um algoritmo genético. Cada cromossomo representa uma possível conformação do *backbone* da proteína. Essa é uma forma padrão de representar uma conformação de uma dada proteína.

Existem diversos métodos para selecionar os cromossomos da população que irão participar do processo de recombinação (cruzamento). Um método popular é conhecido como *seleção proporcional*. Nesse caso um cromossomo é selecionado com um probabilidade $p(x) = u(x)/u(t)$ que é proporcional a razão entre a energia da conformação $x - [u(x)]$ e a energia média das conformações $u(t)$. Isto, no caso do problema de busca conformacional de estruturas terciárias de proteínas.

Os cruzamentos são realizados de diversas formas: dois pontos (*Two-Point*), *Wraparound* e *cruzamento uniforme (Uniform Crossover)*; no caso do problema de predição de estruturas protéicas, o primeiro e o terceiro são os tipos mais utilizados. Na primeira forma, os pares de cromossomos são cortados em um ponto e recombinados de forma que cada cromossomo venha

possuir uma parte que pertencia a ele e uma parte complementar pertencente ao outro cromossomo. No segundo tipo de cruzamento, os pares de cromossomos são divididos em vários pontos e re combinados. (ver Figuras 18 a e b respectivamente).

Cromossomos pais

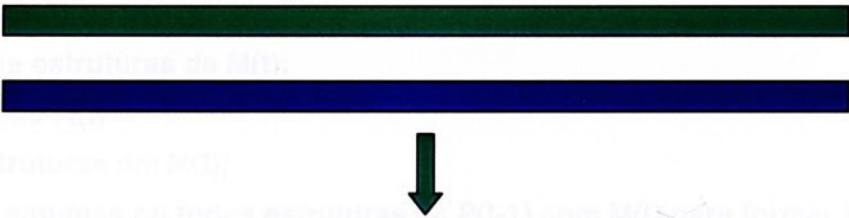


Cromossomos Filhos



FIGURA 18 a – Ilustração de um cruzamento dois pontos

Cromossomos pais



Cromossomos Filhos



FIGURA 18 b – Ilustração de um cruzamento uniforme

É importante lembrar que após o cruzamento o operador de mutação pode ser aplicado nos cromossomos filhos. Esse operador realiza um processo semelhante a mutação genética, alterando alguns genes dos cromossomos filhos (i.e., alterando os valores dos ângulos ϕ e φ no exemplo descrito nessa subseção), como ilustrado na Figura 19.



FIGURA 19. Ilustração do operador de Mutação

De forma genérica, o algoritmo genético opera de acordo com fluxograma montado na figura 20.

Function GA

```
{  
  t=0;  
  inicializa P(t);  
  avalia estruturas em P(t);  
  while (condição de fim não alcançada)  
  {  
    t++;  
    seleciona M(t) de P(t-1);  
    recombina estruturas de M(t);  
    faz mutação M(t)  
    avalia estruturas em M(t);  
    substitui algumas ou todas estruturas de P(t-1) com M(t) para formar P(t);  
  }  
}
```

FIGURA 20 – Pseudocódigo de um exemplo padrão de algoritmo genético.

Estudo do Núcleo Hidrofóbico do Citocromo_{b562}

5.1. Considerações Iniciais

Nesse capítulo é discutida a aplicação de redes neurais do tipo Hopfield e de algoritmos genéticos, em conjunto com biblioteca de rotâmeros, na predição e no design do núcleo hidrofóbico do Citocromo_{b562}. Os resultados desse estudo mostram que ambas as técnicas são eficientes para prever a estrutura de um núcleo hidrofóbico de uma proteína e que com algoritmos genéticos além da predição é possível realizar “design” de núcleos.

Além disso, os estudos comprovam a viabilidade da utilização de bibliotecas de rotâmeros com o objetivo de minimizar o espaço conformacional a ser explorado e reduzir o custo computacional.

Os Citocromos são agrupados em 3 famílias separadas por diferenças nos seus espectros de absorção e denominadas de a, b e c. Os grupos heme dos Citocromos das famílias a e b estão fortemente, mas não covalentemente, ligados às proteínas.

O Citocromo _{b562} estudado pertence a b família e possui 106 aminoácidos. Ele é predominantemente constituído de estruturas secundárias do tipo α -hélice (5 motivos). Essa molécula foi escolhida para estudo para que os resultados obtidos pudessem ser comparados com o trabalho descrito em (Kono e Dói, 1993).



5.2. Biblioteca de Rotâmeros

O empacotamento das cadeias laterais possui um papel importante no enovelamento das proteínas e na estabilidade de suas estruturas terciárias. Consequentemente, elas são de extrema importância para o problema de predição de estrutura (Samudrala et al., 2000).

Parte da dificuldade na construção de modelos de estrutura está na flexibilidade das cadeias laterais dos aminoácidos. Porém, de acordo, com princípios de química estrutural, as várias cadeias laterais exibem preferências conformacionais, ou seja, os ângulos chi's de cada aminoácido possuem valores preferências que podem ser determinados através de um levantamento estatístico das estruturas depositadas nos bancos de dados.

Dunbrack e Karplus discutiram a criação e uso de uma biblioteca de rotâmeros para representar as conformações mais prováveis de cadeias laterais para cada tipo de aminoácido, onde um rotâmero representa uma possível conformação para cadeia lateral de um aminoácido (ver Figura 21a 21 b).

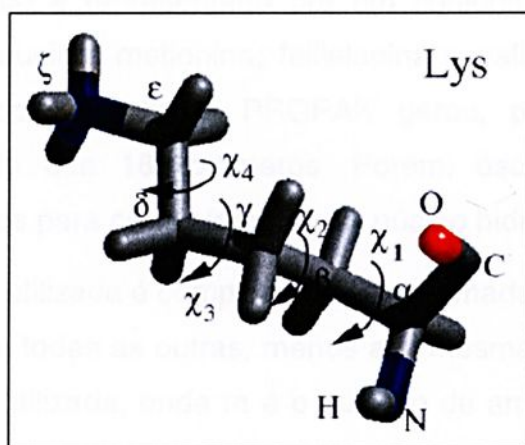


Figura 21- Ilustração da Lysina e dos ângulos de sua cadeia lateral (χ) que dão origem as diferentes conformações desse aminoácido, gerando seus rotâmeros.

Eles forneceram uma lista dos valores mais prováveis para os ângulos chi's, baseado em 2273 estruturas depositadas no banco de dados (*Protein Data Bank*) (Dunbrack and Karplus, 1993, 1994). Neste trabalho, uma biblioteca de

rotâmeros independente do esqueleto foi gerada usando o software PROPAK proposto por Ponder e Richards (Ponder e Richards, 1987). O Propak gera as coordenadas cartesianas de todo os possíveis rotâmeros para cada cadeia lateral de uma molécula.

O objetivo inicial é investigar qual a melhor combinação de rotâmeros para compor o núcleo hidrofóbico do Citocromo_{b562}, restringindo-se a um grupo de seis aminoácidos hidrofóbicos e a uma função Potencial composta apenas por termos do tipo Lennard-Jones.

O núcleo hidrofóbico é formado de 16 resíduo e a base de dados de aminoácidos composta de seis aminoácidos diferentes que podem assumir diferentes estruturas terciárias distintas dando origem aos chamados *rotâmeros*. Dessa forma, o núcleo hidrofóbico possui um espaço conformacional grande.

5.3. Predição do Núcleo Hidrofóbico utilizando Redes de Hopfield

O núcleo hidrofóbico é representado por um conjunto de 6 aminoácidos (alanina, leucina, isoleucina, metionina, felilalanina e valina) e 18 rotâmeros para cada aminoácido. O software PROPAK gerou, para alguns desses aminoácidos, mais do que 18 rotâmeros. Porém, escolheu-se apenas 3 rotâmeros bem distintos para cada elemento do núcleo hidrofóbico.

A rede de Hopfield utilizada é composta de uma camada com $m \times n$ unidades cada qual conectada a todas as outras, menos a si mesma. A Figura 22 ilustra a arquitetura da rede utilizada, onde m é o número de aminoácidos (posições da cadeia da proteínas) que compõem o núcleo hidrofóbico e n representa o número de possíveis rotâmeros para cada posição. Assim, cada configuração da rede neural representa uma conformação da proteína. Ou seja, cada neurônio ativado corresponde a um rotâmero em uma dada posição do núcleo hidrofóbico.



Para cada configuração da rede uma função potencial de energia é calculada. O objetivo do algoritmo de treinamento é encontrar a configuração da RNA que otimize a função potencial definida na Equação 1. O primeiro termo da equação corresponde a soma dos termos de van der Waals entre as cadeias laterais e entre as cadeias laterais e a cadeia principal. O segundo e terceiro termo são necessários para que a rede neural favoreça estados estáveis e evitem que mais de um resíduo esteja ativo para cada posição do núcleo hidrofóbico.

$$E = A/2 \left\{ \sum_{X=1}^m \sum_{Y=1}^m \sum_{i=1}^n \sum_{j=1}^n P_{XiYj}^s V_{Xi} V_{Yj} + \sum_{X=1}^m \sum_{i=1}^n V_{Xi} P_{Xi}^b \right\} + B/2 \left\{ \sum_{X=1}^m \sum_{i=1}^n \sum_{j=1}^n V_{Xi} V_{Xj} \right\} + C/2 \left\{ \sum_{X=1}^m \sum_{i=1}^n V_{Xi} - m \right\}^2$$

Equação 1-Potencial Utilizado pela Rede de Hopfield.

onde A, B e C são constantes definidas experimentalmente para melhorar a convergência da rede neural. Os índices X e Y representam a posição dos resíduos que compõem o núcleo hidrofóbico. Os índices i e j representam os rotâmeros. V_{Xi} representa a saída de um neurônio i para o resíduo na posição X, P_{XiYj} representa o valor da interação de van der Waals entre os rotâmeros i e j dos aminoácidos na posição X e Y (Equação 2). P_{Xi} representa a interação de van der Waals entre o rotâmero i do resíduo na posição X e todo o backbone, m é o número de resíduos que formam o núcleo e n é o número total de rotâmeros possíveis para cada posição.

$$E = \epsilon_{xy}^* \left\{ \left(\frac{r_{xy}^*}{d_{xy}} \right)^{12} - 2 \left(\frac{r_{xy}^*}{d_{xy}} \right)^6 \right\}$$
$$\epsilon_{xy}^* = \sqrt{\epsilon_y^* \cdot \epsilon_x^*}, r_{xy} = r_x + r_y$$

Equação 2- Termo de Interação de van der Waals utilizada pela Rede de Hopfield e pelo algoritmo genético.

onde d_{xy} é a distância entre dois átomos x e y , $\epsilon_{xy} = \sqrt{\epsilon_x \epsilon_y}$ e $r_{xy} = r_x + r_y$. Os parâmetros r_i (raio de van der Waals para o átomo i), ϵ_i (energia para separar um par de átomos do tipo i) utilizados são definidos do campo de força do GROMOS 96.

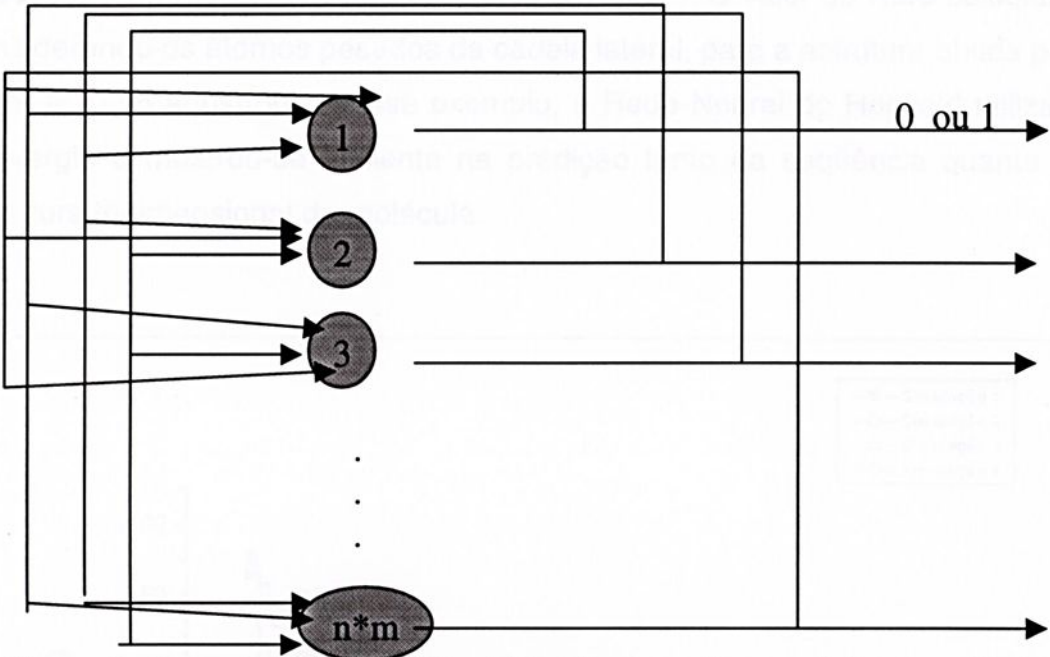


Figura 22 - Ilustração da Rede de Hopfield utilizada, onde o neurônio com saída igual a 1 está ativado e com saída zero desativado.

A Rede de Hopfield, utilizada, apresentou um bom desempenho, predizendo, para o núcleo hidrofóbico, seqüências com energias próximas ou mais baixas do que a energia da estrutura nativa. Na Figura 23, apresenta-se o gráfico da energia das conformações obtidas pela rede neural, em função do número de ciclos de treinamento utilizados em algumas simulações, todas realizadas sobre as mesmas condições iniciais. Pode-se observar pelo gráfico, que a rede convergiu para configurações correspondentes a valores de energia próximos da energia que a estrutura depositada no PDB possui (-179.4 Kca/mol).

Realizando a sobreposição das estruturas de menor energia obtidas com a depositada no PDB, pode-se observar que elas correspondem a estados compactos do núcleo da proteína e apresentam valores de energia mais baixos

do que os encontrados por Kono e Doi (Kono e Doi, 1993). O valor do volume obtido para a região do núcleo hidrofóbico predito ficou muito próximo do valor calculado para a estrutura depositada no PDB. Sendo que o volume do núcleo predito pela rede é 3% maior que o volume do núcleo hidrofóbico da estrutura depositada no PDB (calculado utilizando o *Tinker*). O valor do **RMS** calculado, considerando os átomos pesados da cadeia lateral, para a estrutura obtida pela rede é 0.96 angstroms. Nesse exemplo, a Rede Neural de Hopfield utilizada convergiu e mostrou-se eficiente na predição tanto da sequência quanto da estrutura tridimensional da molécula.

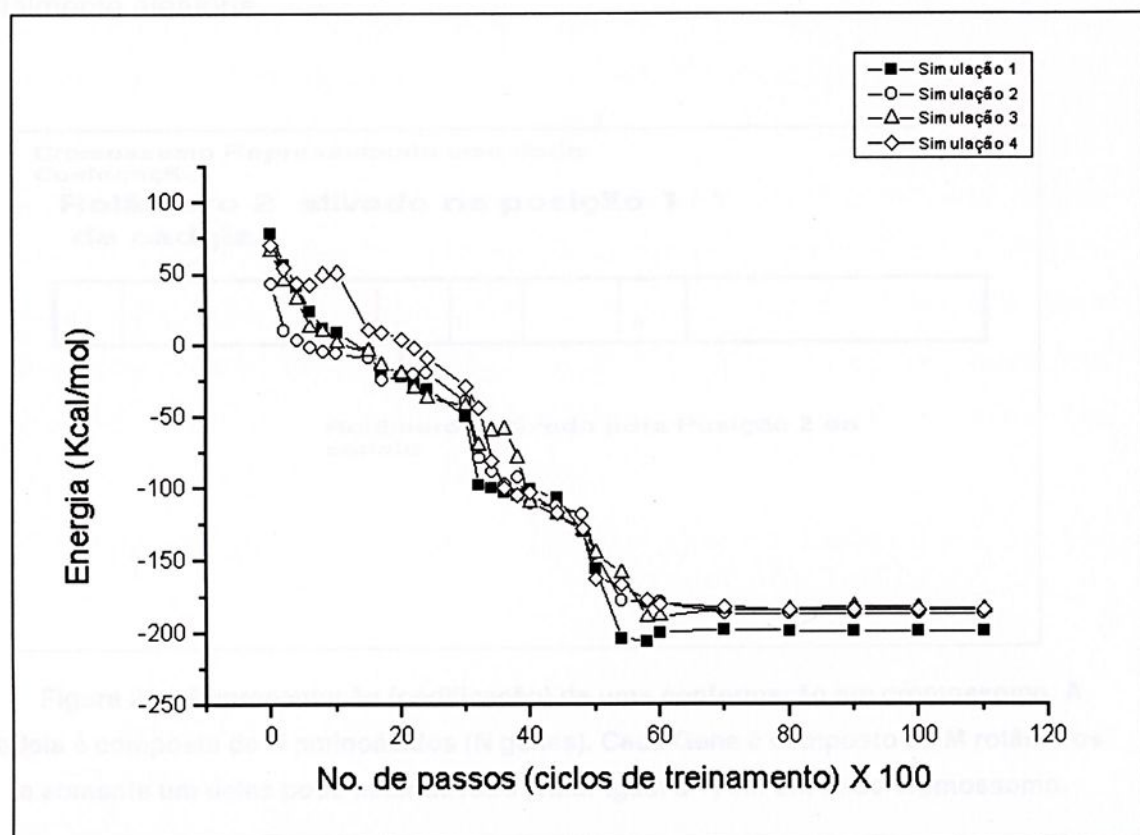


Figura 23– O Gráfico apresenta a energia das estruturas mais compactadas em função do número de passos da rede neural. Partindo-se de estruturas (do núcleo hidrofóbico) de alta energia (acima de 50 Kcal/mol); a rede faz a energia convergir para o patamar em torno de -190 Kcal/mol.

dois filhos que são aceitos segundo o critério de energia (Metrópolis) (Rosenbluth, 1953). Se a média da energia dos filhos for menor que a média dos pais, os filhos são aceitos automaticamente.

Caso contrário, um peso de Boltzmann (critério de metrópolis) é utilizado para decidir a aceitação ou não dos filhos. Podem ocorrer diferentes situações: *um único filho ser rejeitado e um pai replicado para geração seguinte, ou ambos os filhos serem aceitos ou ainda ambos os filhos serem rejeitados e os pais copiados para a geração seguinte*. No caso do operador de mutação, a escolha dos cromossomos que sofrem mutação é aleatória bem como a escolha da posição da mutação.

O tamanho da população inicial, o número de gerações, quantidade de posições (aminoácidos) do caroço e a taxa de mutação são parâmetros ajustados pelo usuário para cada simulação.

Os átomos que compõem o sistema interagem através de uma função potencial do tipo Lennard-Jones (Equação 2). No caso do algoritmo genético, a soma restringe-se às interações entre átomos das cadeias laterais e entre átomos das cadeias laterais e átomo da cadeia principal. Interações entre dois átomos da cadeia principal não são consideradas porque ela está fixa forneceria sempre a mesma constante fixa.

O programa, que implementa o algoritmo genético, possui duas versões diferentes:

(i) na primeira versão, a estrutura da cadeia principal e a seqüência de aminoácidos do núcleo hidrofóbico são fixas e apenas os rotâmeros, em cada posição, são trocados;

(ii) na segunda versão, as coordenadas da cadeia principal são mantidas fixas e a seqüência de aminoácidos e os rotâmeros são trocados. Dessa forma, podia-se projetar novas seqüências para o núcleo hidrofóbico da proteína através do algoritmo genético.

O programa possui 5 arquivos de dados e entrada: *parametros.dat*, *rotamer.dat* e *backbone.dat*, *wals.dat* e o arquivo *PDB da proteína*. O primeiro



arquivo permite o usuário parametrizar sua simulação. Entre outras coisas, através desse arquivo é possível: (i) definir se deseja realizar predição ou design (i.e. manter ou variar a seqüência de aminoácidos do núcleo); (ii) definir o tamanho da população, a porcentagem de cromossomos que são copiados diretamente de uma geração para outra; (iii) definir a taxa de mutação; (iv) definir o número máximo de gerações; (v) especificar as posições na seqüência que constituem o núcleo; (vi) definir o número de resíduos do núcleo hidrofóbico.

O arquivo 'rotamer.dat' contém as coordenadas cartesianas do conjunto de rotâmeros que será utilizado pelo programa. Por fim, o arquivo 'backbone.dat' contém as coordenadas cartesianas do backbone extraídas, automaticamente, do arquivo PDB fornecido pelo usuário. Esse arquivo, ao contrário do 'rotamer.dat', é gerado automaticamente sem intervenção manual do usuário para editar o mesmo. Por último, o arquivo 'wals.dat' contém os parâmetros de van der Waals para cada tipo de átomo do PDB.

O programa gera como saída os arquivos PDBs das estruturas que possuírem um valor de energia menor que a da estrutura nativa acrescida de 10% desse valor.

O sistema foi testado para prever o núcleo hidrofóbico do **Citocromo_{b562}**, mantendo a seqüência fixa (a menos das posições 98 e 101 que tiveram seus aminoácidos trocados por aminoácidos apolares). Nesse primeiro estudo, foram obtidas estruturas com energias próximas à nativa.

O RMSD calculado entre a superposição dessas estruturas preditas, de menor energia, com a nativa variaram entre 0.7 Angstroms e 2.4 Angstroms. Na Figura 25, a energia média por geração em função do número de gerações é graficada. Nota-se que após 40 gerações esse valor flutua em torno de um patamar de energia de -155 Kcal/mol. Na Figura 26, o mesmo tipo de gráfico foi feito para a energia mínima por geração. O patamar corresponde a uma energia de -188 Kcal/mol, que é menor que a nativa -179 kcal/mol. Pode-se notar que alguns mínimos locais foram encontrados após 10, 20 e 30 gerações.



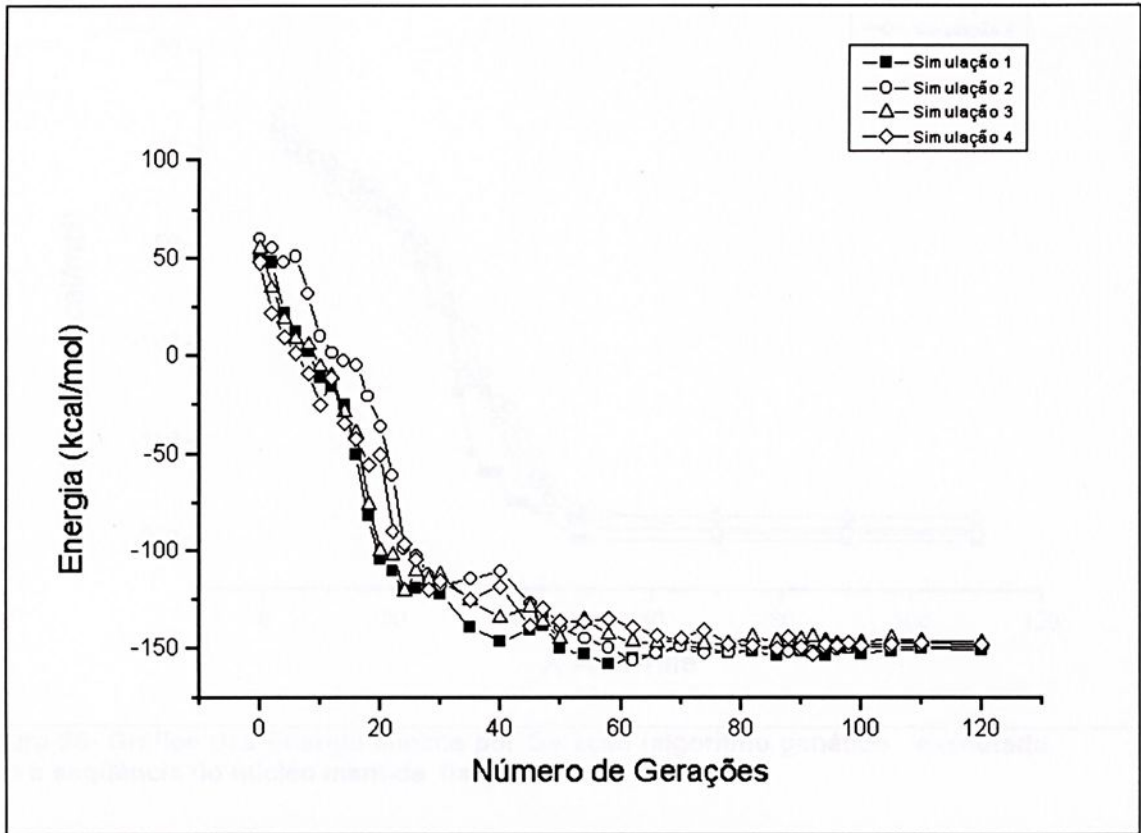


Figura 25- Gráfico das Energia Média das Conformações por Geração (algoritmo genético executado com a seqüência do core fixa).

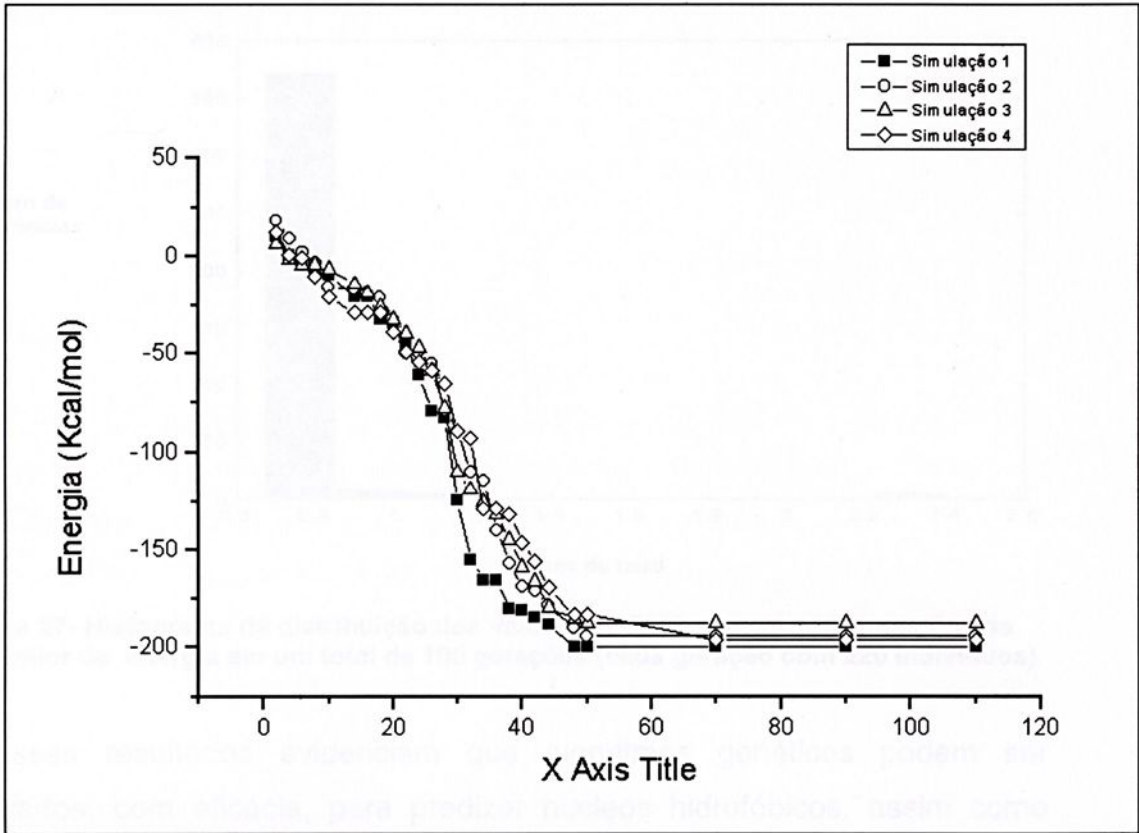


Figura 26- Gráfico das Energia Mínima por Geração (algoritmo genético executado com a sequência do núcleo mantida fixa).

Na Figura 27, está apresentado o histograma do rmsd dos átomos pesados das cadeias laterais das estruturas preditas de menor energia. Pode-se notar que para as estruturas de energia baixa, o rmsd concentra-se na região entre 0,6 e 1,2 angstroms.

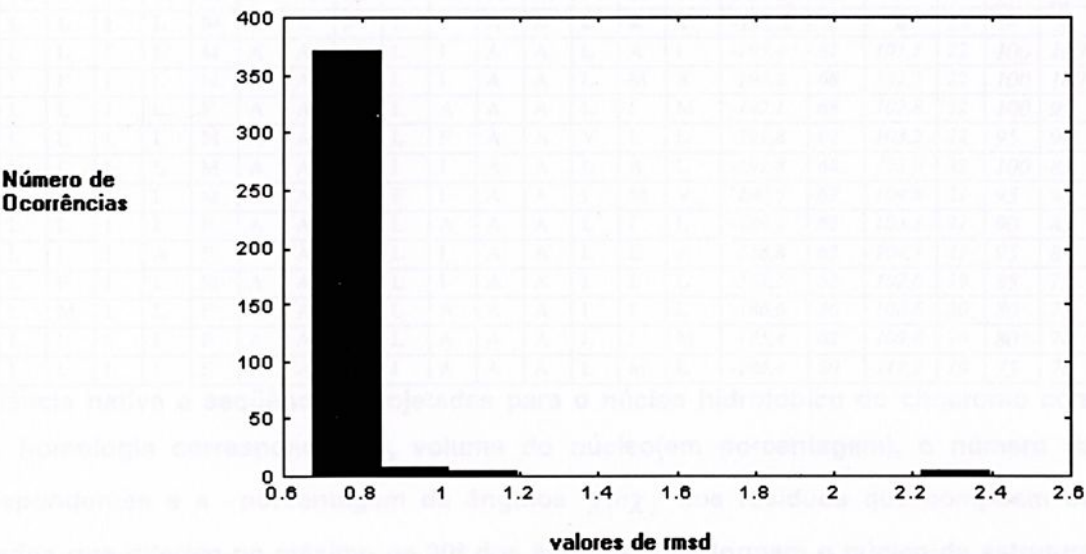


Figura 27- Histograma da distribuição dos valores de RMS para as conformações de menor valor de energia em um total de 100 gerações (cada geração com 220 indivíduos).

Esses resultados evidenciam que algoritmos genéticos podem ser utilizados, com eficácia, para predizer núcleos hidrofóbicos, assim como descrito com redes de Hopfield.

O algoritmo também foi testado para projetar um novo núcleo hidrofóbico para o Citocromo B₅₆₂. Nesse caso, foi mantido fixo a Metionina na posição 7, como na estrutura nativa, enquanto nas outras posições, do núcleo hidrofóbico, foram variados os tipos de aminoácidos e os rotâmeros. Foram realizadas várias simulações com o tamanho da população inicial variando entre 200 e 200000 indivíduos (configurações).

a)População com 200 conformações

Nesse caso, obteve-se estruturas com uma seqüência primária para núcleo hidrofóbico com até 81% de homologia em relação à seqüência depositada no PDB, porém com energias da mesma ordem ou bem próximas da nativa e com um número de contatos hidrofóbico próximos ao número de contatos da estrutura depositada no PDB. Mostrado na Tabela 1 , nas quais estão apresentadas a seqüência, o volume, a energia e o número de contatos do núcleo da estrutura nativa e das dez estruturas de energia baixa.

7	10	14	17	30	33	37	40	65	68	72	75	91	94	98	101	Energy	Id	Vol	Co	χ_1	χ_2
M	L	L	I	L	M	A	A	F	L	I	A	A	L	R	Y	-191,2	%	%	24	%	%
M	L	L	I	L	M	A	A	A	L	I	A	A	L	A	I	-193,4	81	101,5	22	100	100
M	I	F	I	L	M	A	A	A	L	I	A	A	L	M	A	-193,2	68	102,2	22	100	100
M	L	L	I	L	F	A	A	L	L	A	A	A	L	I	M	-192,1	68	102,8	22	100	95
M	L	L	L	I	M	A	A	I	L	F	A	A	V	I	L	-191,8	62	103,2	22	95	90
M	L	L	I	L	M	A	A	L	I	I	A	A	L	A	L	-191,8	68	103,9	22	100	83
M	L	I	I	L	M	A	A	V	F	I	A	A	I	M	V	-190,5	62	104%	21	95	90
M	L	L	I	I	F	A	A	L	L	A	A	A	L	I	L	-189,2	62	105,3	21	90	85
M	L	I	I	A	F	A	A	L	L	I	A	A	L	L	A	-188,8	62	104,3	21	95	85
M	L	F	L	L	M	A	A	I	L	I	A	A	I	L	L	-187,5	62	107,6	19	85	75
M	L	M	L	L	F	A	A	F	L	A	A	A	I	I	L	-186,6	56	108,6	20	80	75
M	I	L	I	I	F	A	A	F	L	A	A	A	L	I	M	-185,4	62	109,8	20	80	70
M	I	L	I	I	F	A	A	L	I	A	A	A	L	M	L	-186,4	50	112,2	19	75	70

Tabela 1- Sequência nativa e seqüências projetadas para o núcleo hidrofóbico do citocromo com suas energias, homologia correspondentes, volume do núcleo(em porcentagem), o número de contato correspondentes e a porcentagem de ângulos χ_1 e χ_2 dos resíduos que compõem os núcleos projetados que diferem no máximo de 20º dos ângulos que formam o núcleo da estrutura depositada no PDB.

Com os resultados apresentados pode-se propor as seqüências de número 1 e 2 como possíveis novos núcleos hidrofóbicos para o citocromo. E podemos notar que o nosso método obtém uma varredura do espaço conformacional tão boa quanto proposto em (Kono e Doi, 1993) que utiliza autômato celular.

b)População com 2000 conformações

Nessas simulações, obteve-se um maior número de estruturas com uma seqüência primária para núcleo hidrofóbico com até 81% de homologia em relação à seqüência depositada no PDB, porém com energias da mesma ordem ou bem próximas da nativa e com um número de contatos hidrofóbico próximos ao número de contatos da estrutura depositada no PDB. Mostrado na Tabela 2.

7	10	14	17	30	33	37	40	65	68	72	75	91	94	98	101	Energy	Id	Vol	Co	χ_1	χ_2
M	L	L	I	L	M	A	A	F	L	I	A	A	L	R	Y	-191,2	%	%	24	%	%
M	L	L	I	L	M	A	A	L	L	I	A	A	L	V	M	-193,8	81	101,2	22	100	100
M	L	I	I	L	M	A	A	F	L	I	A	A	L	M	L	-192,8	81	101,8	22	100	100
M	L	L	I	L	M	A	A	F	I	V	A	A	L	V	L	-192,4	75	102,4	22	100	100
M	L	L	L	L	M	A	A	F	L	I	A	A	L	I	L	-191,8	81	102,8	21	100	95
M	I	L	I	L	M	A	A	F	L	I	A	A	L	V	V	-191,5	81	103,4	22	100	90
M	L	V	I	L	M	A	A	V	L	I	A	A	L	M	L	-191,2	75	103,6	21	95	95
M	L	L	I	I	F	A	A	L	L	I	A	A	L	I	L	-190,6	68	103,8	21	90	85
M	L	L	I	L	V	A	A	F	L	V	A	A	L	V	M	-190,5	75	104,1	21	100	85
M	L	L	L	L	M	A	A	I	L	I	A	A	L	M	L	-190,0	75	104,4	21	100	90
M	L	V	L	L	M	A	A	L	L	I	A	A	L	L	L	-189,8	68	104,6	20	90	75
M	L	L	I	I	M	A	A	V	L	I	A	A	L	V	L	-189,6	75	104,8	20	90	85
M	L	L	I	I	L	A	A	F	L	L	A	A	I	V	M	-188	68	106	19	80	70

Tabela 2- Sequência nativa e seqüências projetadas para o núcleo hidrofóbico do citocromo com suas energias, homologia correspondentes, volume do núcleo(em porcentagem), o número de contato correspondentes e a porcentagem de ângulos χ_1 e χ_2 dos resíduos que compõem os núcleos projetados que diferem no máximo de 20° dos ângulos que formam o núcleo da estrutura depositada no PDB.

b)População com 20000 conformações

No caso das simulações com uma população fixa de 20000 indivíduos (configurações), foram obtidas 5 conformações com homologia de até 81% com a depositada no PDB e 7 conformações com energia menor do que nativa, como pode-se observar na Tabela 3.

7	10	14	17	30	33	37	40	65	68	72	75	91	94	98	101	Energy	Id	Vol	Co	χ_1	χ_2
M	L	L	I	L	M	A	A	F	L	I	A	A	L	R	Y	-191,2	%	%	24	%	%
M	L	I	I	L	M	A	A	F	L	I	A	A	L	L	M	-194,2	81	100,6	22	100	100
M	L	V	I	L	M	A	A	F	L	I	A	A	L	V	L	-193,8	81	101,2	22	100	100
M	L	L	I	L	M	A	A	F	L	V	A	A	L	L	V	-193,2	81	101,5	22	100	100
M	L	V	I	L	M	A	A	F	L	I	A	A	L	M	V	-192,8	81	102,2	22	100	100
M	L	L	L	L	M	A	A	F	L	I	A	A	L	L	L	-192,5	81	102,7	21	100	95
M	L	L	I	V	M	A	A	F	I	I	A	A	L	M	L	-192	75	103,0	21	95	95
M	L	L	I	L	M	A	A	L	L	I	A	A	V	I	L	-191,6	75	103,4	21	95	85
M	L	L	I	L	M	A	A	I	L	I	A	A	V	L	L	-191,2	75	103,6	21	100	85
M	L	I	I	L	M	A	A	V	L	V	A	A	L	L	M	-190,8	68	103,8	21	100	80
M	L	V	I	L	M	A	A	L	I	I	A	A	L	L	M	-190,4	68	104,0	20	100	75
M	L	L	I	L	M	A	A	V	L	L	A	A	L	V	V	-189,6	75	104,1	20	95	75
M	L	L	I	I	M	A	A	L	L	L	A	A	I	V	M	-189,2	62	104,6	19	85	75

Tabela 3- Sequência nativa e seqüências projetadas para o núcleo hidrofóbico do citocromo com suas energias, homologia correspondentes, volume do núcleo(em porcentagem), o número de contato correspondentes e a porcentagem de ângulos χ_1 e χ_2 dos resíduos que compõem os núcleos projetados que diferem no máximo de 20° dos ângulos que formam o núcleo da estrutura depositada no PDB.

b)População com 200000 conformações

Nessas simulações, obteve-se 7 conformações com homologia de 81% em relação à seqüência depositada no PDB, porém com energias da mesma ordem ou bem próximas da nativa e com um número de contatos hidrofóbico próximos ao número de contatos da estrutura depositada no PDB. Mostrado na Tabela 4.

7	10	14	17	30	33	37	40	65	68	72	75	91	94	98	101	Energy	Id	Vol	Co	χ_1	χ_2
M	L	L	I	L	M	A	A	F	L	I	A	A	L	R	Y	-191,2	%	%	24	%	%
M	L	I	I	L	M	A	A	F	L	I	A	A	L	I	M	-194,6	81	100,8	22	100	100
M	L	V	I	L	M	A	A	F	L	I	A	A	L	V	M	-194,0	81	101,4	22	100	100
M	L	L	I	V	M	A	A	F	L	I	A	A	L	L	L	-193,6	81	101,9	22	100	100
M	L	L	V	L	M	A	A	F	L	I	A	A	L	V	V	-193,1	81	102,4	22	100	100
M	L	L	V	L	M	A	A	F	L	I	A	A	L	L	V	-192,7	81	102,8	22	100	100
M	L	L	I	L	M	A	A	V	L	I	A	A	L	I	L	-192,2	81	103,3	22	100	95
M	L	L	I	L	M	A	A	L	L	I	A	A	L	M	L	-191,8	81	103,6	21	95	95
M	L	I	V	L	M	A	A	F	L	V	A	A	L	L	I	-191,4	75	104,4	21	100	85
M	L	L	I	L	M	A	A	F	V	I	A	A	V	L	V	-191,0	75	104,8	21	100	85
M	L	V	I	L	M	A	A	L	I	I	A	A	L	L	V	-190,6	68	104,6	20	100	75
M	L	L	I	L	M	A	A	V	L	L	A	A	L	L	L	-190,0	75	104,3	20	95	75
M	L	V	L	L	M	A	A	F	L	L	A	A	I	V	L	-189,4	62	105,5	19	85	75

Tabela 4- Seqüência nativa e seqüências projetadas para o núcleo hidrofóbico do citocromo com suas energias, homologia correspondentes, volume do núcleo(em porcentagem), o número de contato correspondentes e a porcentagem de ângulos χ_1 e χ_2 dos resíduos que compõem os núcleos projetados que diferem no máximo de 20º dos ângulos que formam o núcleo da estrutura depositada no PDB.

Analisando os resultados das diferentes simulações com tamanhos de população diferentes, pode-se concluir que o aumento da população permite que o algoritmo genético faça uma varredura melhor do espaço conformacional definido pela base de dados de rotâmeros. Ou seja, há uma relação entre o tamanho da população e o desempenho do algoritmo genético quando utilizamos biblioteca de rotâmeros. As simulações foram realizadas em um Pentium III 900 MHz e em um Pentium IV 2.4 GHz, o tempo de processamento do algoritmo possui uma dependência com o tamanho da população da ordem de n , aproximadamente. O problema maior em relação ao tempo de processamento é o tamanho dos cromossomos, já que nesse caso a dependência é em termos de n^2 , o que limita o tamanho dos peptídeos. Para uma população de tamanho 200, o algoritmo genético gasta cerca de 20 minutos (sendo executado em um Pentium III).

Algoritmos Genéticos Aplicados na Predição de Estrutura de Peptídeos

6.1.Considerações Iniciais

Nesse capítulo discute-se a utilização do algoritmo genético acoplado ao uso de biblioteca de rotâmeros na predição da estrutura de peptídeos. Foram realizados testes com a biblioteca independente da cadeia principal, no sentido descrito no capítulo anterior, e testes com a biblioteca de rotâmeros dependente da cadeia principal. Os primeiros testes foram aplicados na predição da estrutura do Mastoporan-X e nesse caso três funções potenciais foram utilizadas.

No segundo caso, o algoritmo foi aplicado na predição da estrutura da polialanina, esse caso constituiu um teste para a cadeia principal, pois a polialanina tem um único rotâmero como cadeia lateral, e da poliisoleucina.

6.2.Predição do Mastoporan-X

Esse peptídeo é extraído da Vespa Xanthoptera, possui 15 resíduos, sendo que um resíduo é um heteroátomo (grupo NH₂ terminal). Sua seqüência é *INWKGIAAMAKLLX*. A estrutura dessa molécula foi resolvida por Ressonância Magnética Nuclear (RMN) e depositada no PDB (Kusunoki et al.,

1998). O Mastoproan-X apresenta uma estrutura de α -hélice anfipática distorcida.

6.2.1. Predição do Mastoporan-X com Cadeia Principal Fixa

O algoritmo genético implementado para a predição da estrutura do peptídeo é muito semelhante ao implementado para estudar o núcleo hidrofóbico (ver Capítulo 5). Ou seja, a filosofia e os parâmetros de controle são os mesmos. As modificações estão relacionadas às funções de energia utilizadas, à sua parametrização e cálculo.

Nesse algoritmo, as coordenadas da cadeia principal foram fixadas nas posições nativas e uma biblioteca de rotâmeros não dependente do backbone foi utilizada (Ponder & Richards, 1987) para predição e projeção de estruturas/seqüências desse peptídeo. A função e os parâmetros utilizados foram os mesmos descritos no capítulo anterior.

Esse algoritmo também foi implementado com duas opções: (i) manter a seqüência de aminoácidos fixa, isto é, realizar a predição da estrutura terciária baseada na seqüência e na biblioteca de rotâmeros; (ii) e permitir trocar tanto a seqüência de aminoácidos como de rotâmeros, de forma a permitir o "design" de uma nova seqüência para uma dada estrutura conhecida.

a) van der waals

Foram realizadas diversas predições do Mastoporan-X utilizando o algoritmo descrito anteriormente. Nesse conjunto de dados a função energia utilizada continha somente o termo de van der waals, como descrito no Capítulo 5. Na Figura 28 está disposta a energia média por geração em função do número de gerações. Pode-se observar que essa energia converge para um patamar estável, ao redor de -6 Kcal/mol, após 35 gerações.

Na Figura 29, é apresentado o gráfico da menor energia obtida por geração como função do número de gerações. Pode-se notar que o algoritmo fica preso em alguns mínimos locais entre 10 e 20 gerações e após 30 gerações converge para um mínimo de energia de -6.8 Kcal/mol e permanece estável por mais de 100 gerações.

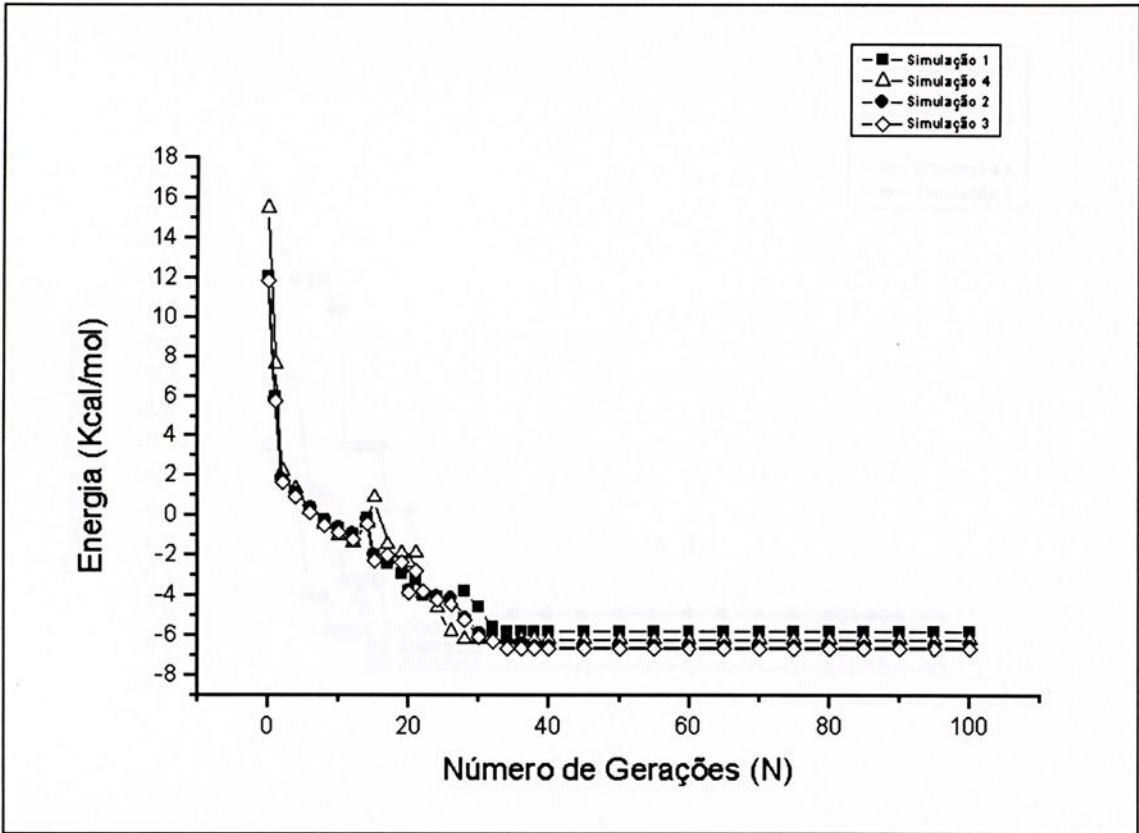


Figura 28- Energia Média por Geração como Função do número de gerações.

Pode-se observar nos gráficos das Figuras 28 e 29 que as energias médias iniciais são diferentes. Porém, as energias médias e mínimas finais convergem para um mesmo patamar. Além disso, todas as simulações realizadas apresentam um comportamento semelhante. Ainda deve-se observar que as energias médias convergem para valores muito próximos nas diferentes simulações.

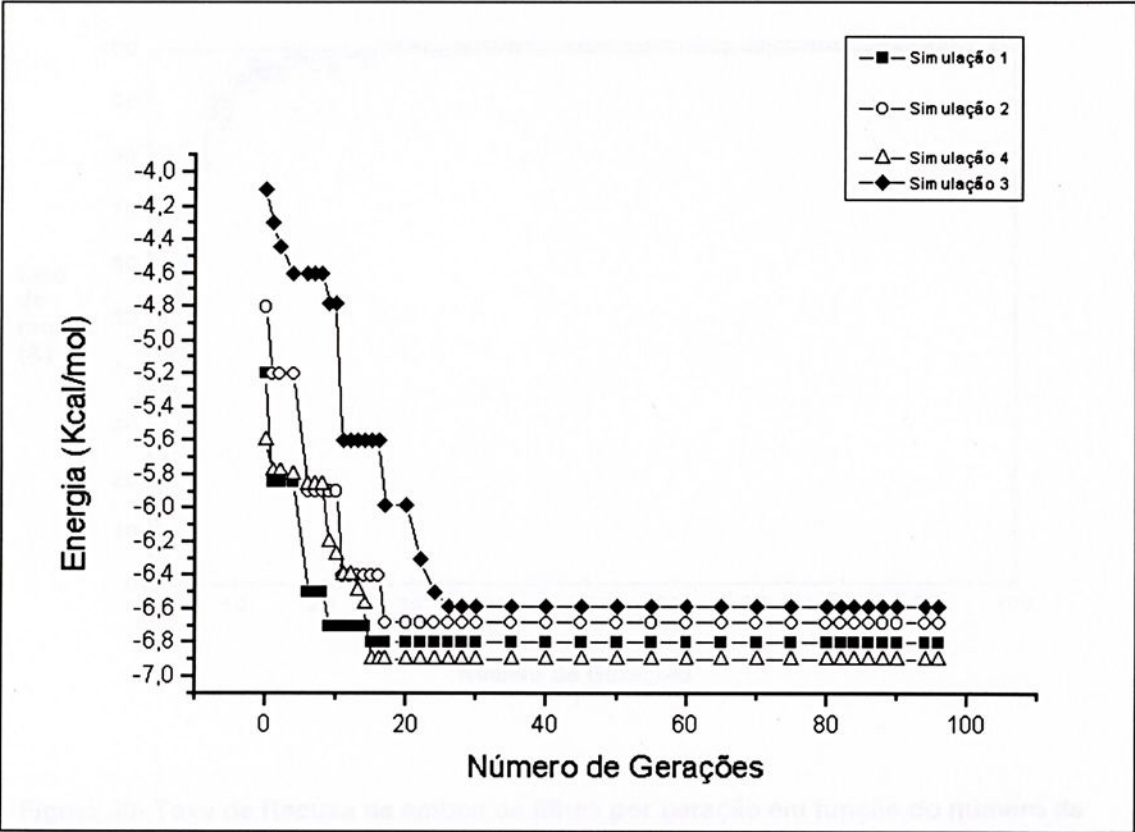


Figura 29- Energia mínima por geração como função do número de gerações.

Na Figura 30 está apresentada a taxa de recusa de ambos os filhos gerados pelo cruzamento por geração em função do número de gerações. Pode-se observar que após 30 gerações, essa taxa atinge praticamente 100%, ou seja, nenhuma nova conformação é aceita. Deve-se observar que por volta da trigésima geração a energia mínima estabiliza e não se altera mais.

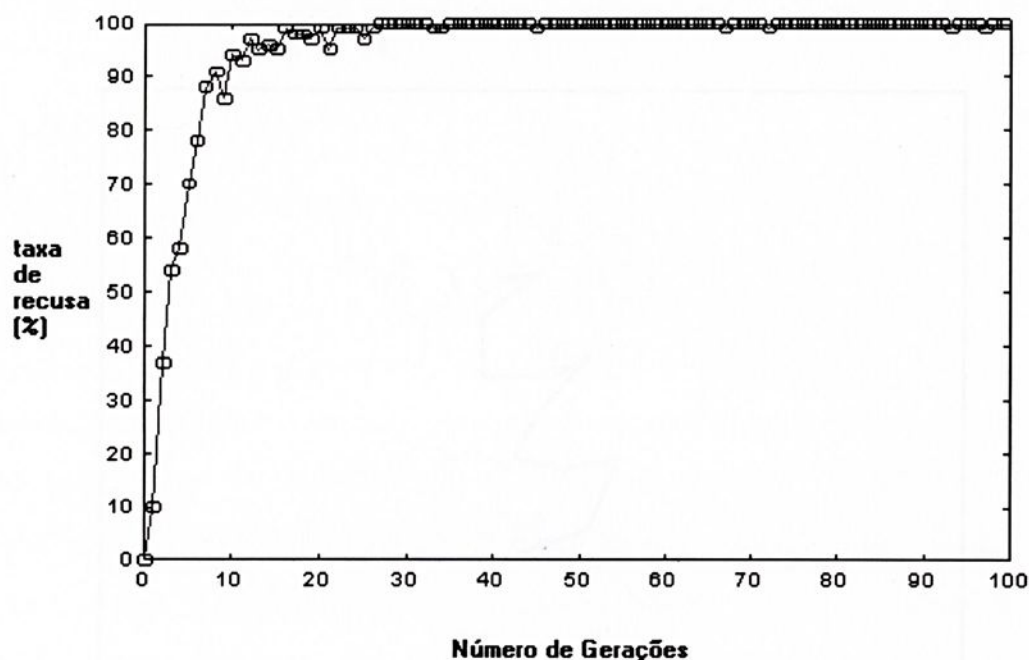


Figura 30- Taxa de Recusa de ambos os filhos por geração em função do número de gerações.

Também foi calculado o rmsd dos átomos pesados da cadeia lateral, entre as conformações, de menor energia, obtidas e a depositada no PDB. Os valores de rmsd variaram entre 0.98 e 1.01 angstroms. Além disso, também, foi realizada a sobreposição das quatro estruturas, de mais baixa energia com a estrutura nativa. O resíduo que apresenta maior diferença na sobreposição é o Triptofano, que, em todas conformações obtidas pelo algoritmo está mais exposto ao solvente, o que indica um resultado não muito bom (ver Figura 31).

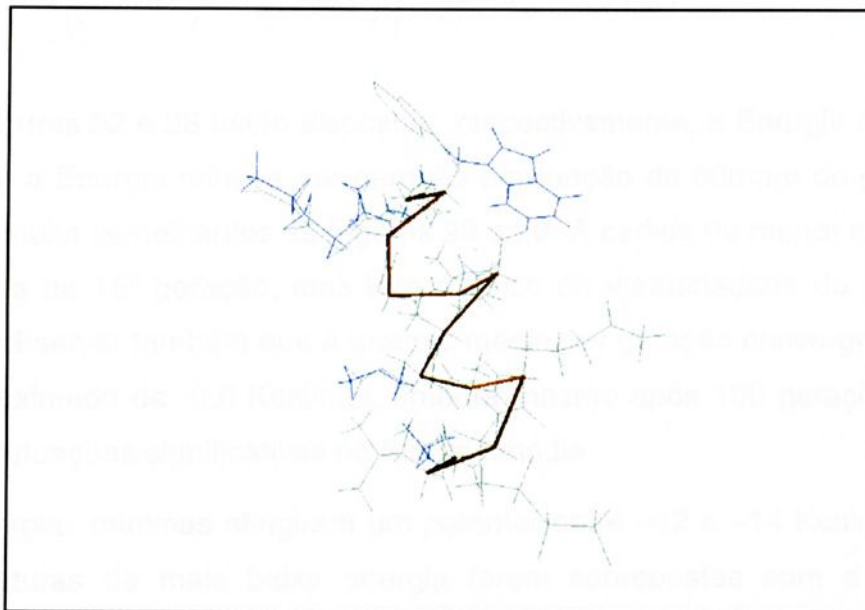


Figura 31– Sobreposição da estrutura depositada em PDB (azul) e a estrutura de menor energia (verde) obtida. Pode-se notar que o Triptofano possui uma posição significativamente diferente nas duas estruturas (nativa e obtida). As demais cadeias laterais mostram uma boa concordância.

b) Introdução do potencial coulombiano

Uma possível razão para os resultados obtidos acima pode estar na simplicidade do potencial de interação adotado. Considerando a importância das interações eletrostáticas e do fato da conformação depositada no PDB do peptídeo em estudo ser uma hélice anfipática, optamos por introduzir um termo para descrever as interações eletrostáticas.

A equação abaixo representa a função potencial utilizada. Ela é constituída do termo de van der Waals, como descrito no Capítulo 5, e do termo coulombiano. As cargas parciais (q_x e q_y) utilizadas foram as mesmas do GROMOS e o solvente foi tratado como um meio contínuo, caracterizado por uma constante dielétrica dependente da distância ($\epsilon = d_{xy}$).

$$E = \varepsilon_{xy}^* \left\{ \left(\frac{r_{xy}^*}{d_{xy}} \right)^{12} - 2 \left(\frac{r_{xy}^*}{d_{xy}} \right)^6 \right\} + \sum_{x,y} \frac{1}{d_{xy}} \left\{ \frac{q_x q_y}{d_{xy}} \right\}$$

Nas Figuras 32 e 33 estão dispostas, respectivamente, a Energia média por geração e a Energia mínima por geração em função do número de gerações. Elas são muito semelhantes às Figuras 28 e 29. A cadeia de menor energia foi encontrada na 15ª geração, mas isso é típico da aleatoriedade do processo. Pode-se observar também que a energia média por geração converge para um valor aproximado de -9,0 Kcal/mol, embora mesmo após 100 gerações ainda existam flutuações significativas na Energia média.

As energias mínimas atingiram um patamar entre -12 e -14 Kcal/mol. Três das estruturas de mais baixa energia foram sobrepostas com a nativa e apresentaram muito pouca diferença em relação às estruturas obtidas nas situações utilizando somente o termo do tipo Lennard-Jones como função potencial

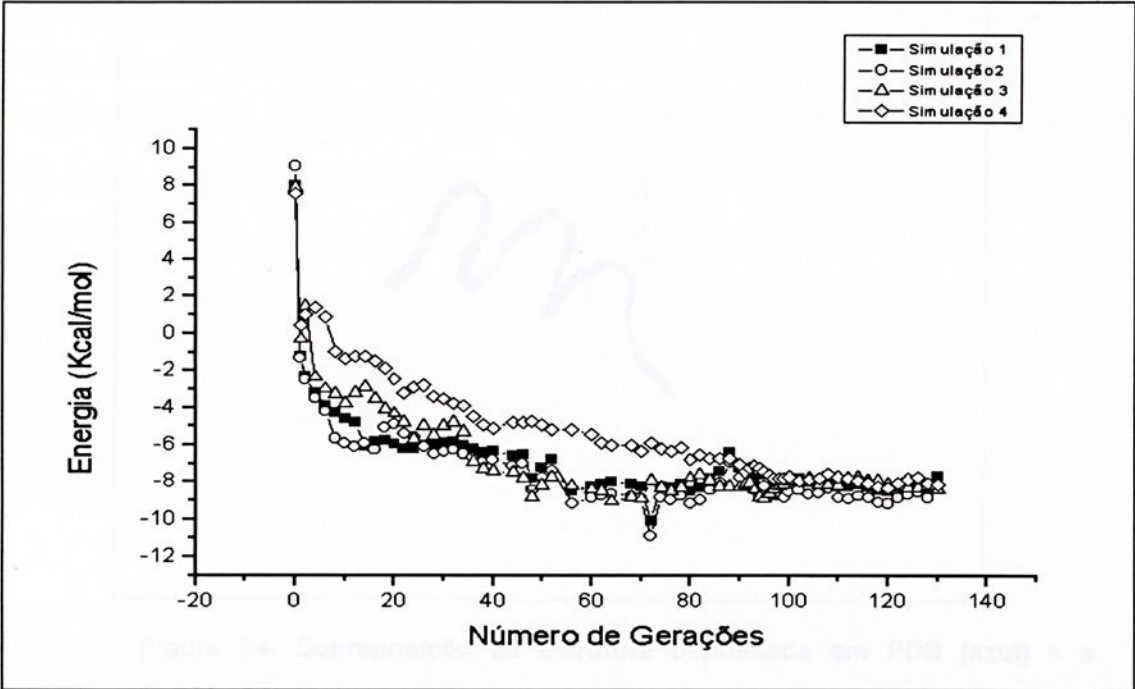


Figura 32- Energia Média por Geração com um Campo de Força Composto por um Termo do Tipo Lennard Jones e um Termo Eletrostático.

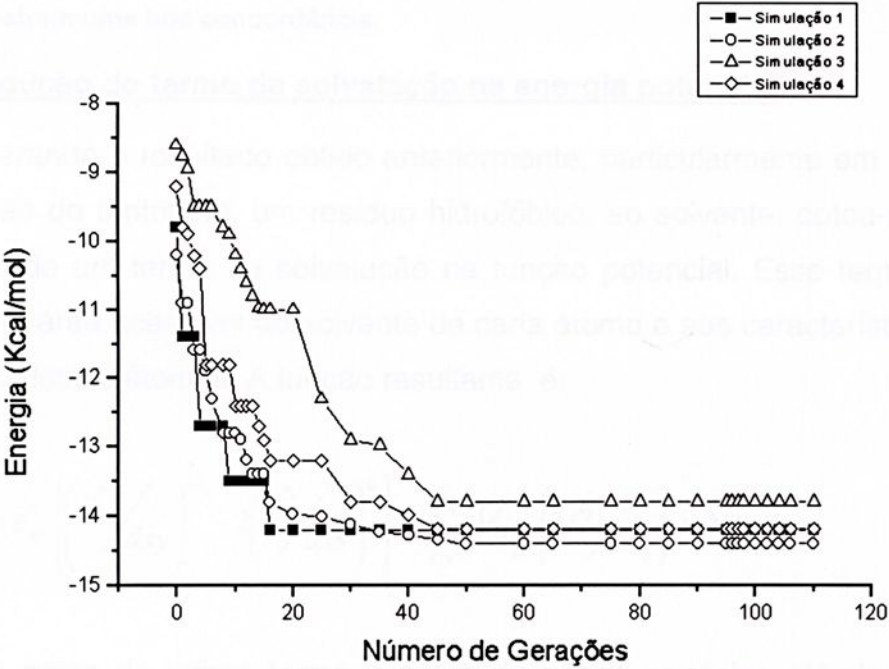


Figura 33- Energia Mínima por Geração com um Campo de Força Composto por um Termo do Tipo Leonard Jones e um Termo Eletrostático.

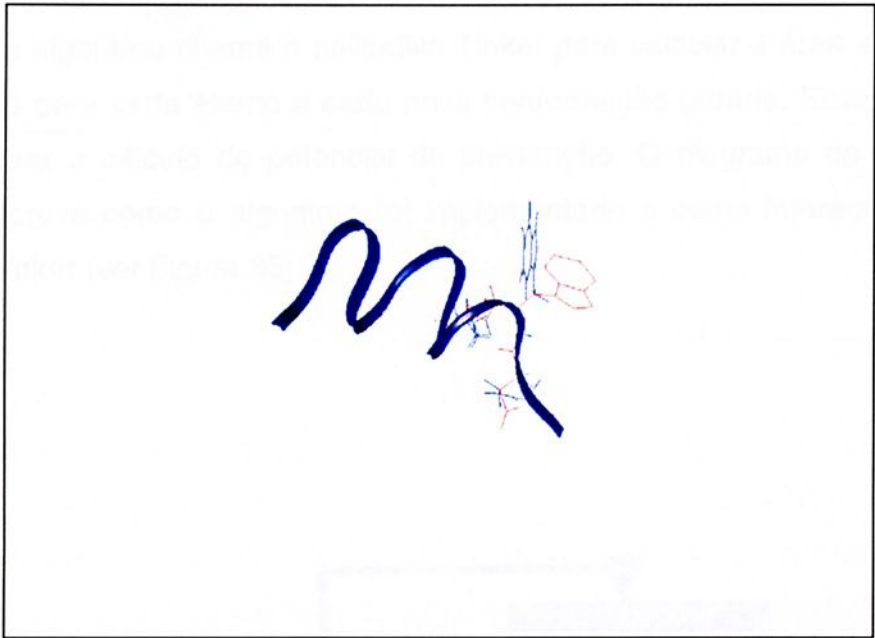


Figura 34- Sobreposição da estrutura depositada em PDB (azul) e a estrutura de menor energia (vermelho) obtida, utilizando um termo eletrostático e um termo do tipo Leonard Jones. Pode-se notar que o Triptofano possui uma posição significativamente diferente nas duas estruturas (nativa e obtida). As demais cadeias laterais mostram uma boa concordância.

c)Introdução do termo de solvatação na energia potencial

Considerando o resultado obtido anteriormente, particularmente em relação à exposição do triptofano, um resíduo hidrofóbico, ao solvente, optou-se pela introdução de um termo de solvatação na função potencial. Esse termo está baseado na área acessível ao solvente de cada átomo e nas características de solvatação destes átomos. A função resultante é:

$$E = \epsilon_{xy}^* \left\{ \left(\frac{r_{xy}^*}{d_{xy}} \right)^{12} - 2 \left(\frac{r_{xy}^*}{d_{xy}} \right)^6 \right\} + \sum_{x,y} \frac{1}{d_{xy}} \left\{ \frac{q_x q_y}{d_{xy}} \right\} + \sum_i \gamma_i A_i$$

Onde a soma do último termo é sobre as várias espécies atômicas e A_i representa a área acessível ao solvente de todos os átomos daquela espécie na molécula. Os parâmetros γ_i são aqueles de Wang e colaboradores (Wang et al., 2001).

O algoritmo precisou ser adaptado em relação ao descrito anteriormente. Para isso o algoritmo chama o aplicativo Tinker para calcular a área acessível ao solvente para cada átomo a cada nova conformação gerada. Essa área é utilizada para o cálculo do potencial de solvatação. O diagrama de Bloco a seguir descreve como o algoritmo foi implementado e como interage com o software Tinker (ver Figura 35).

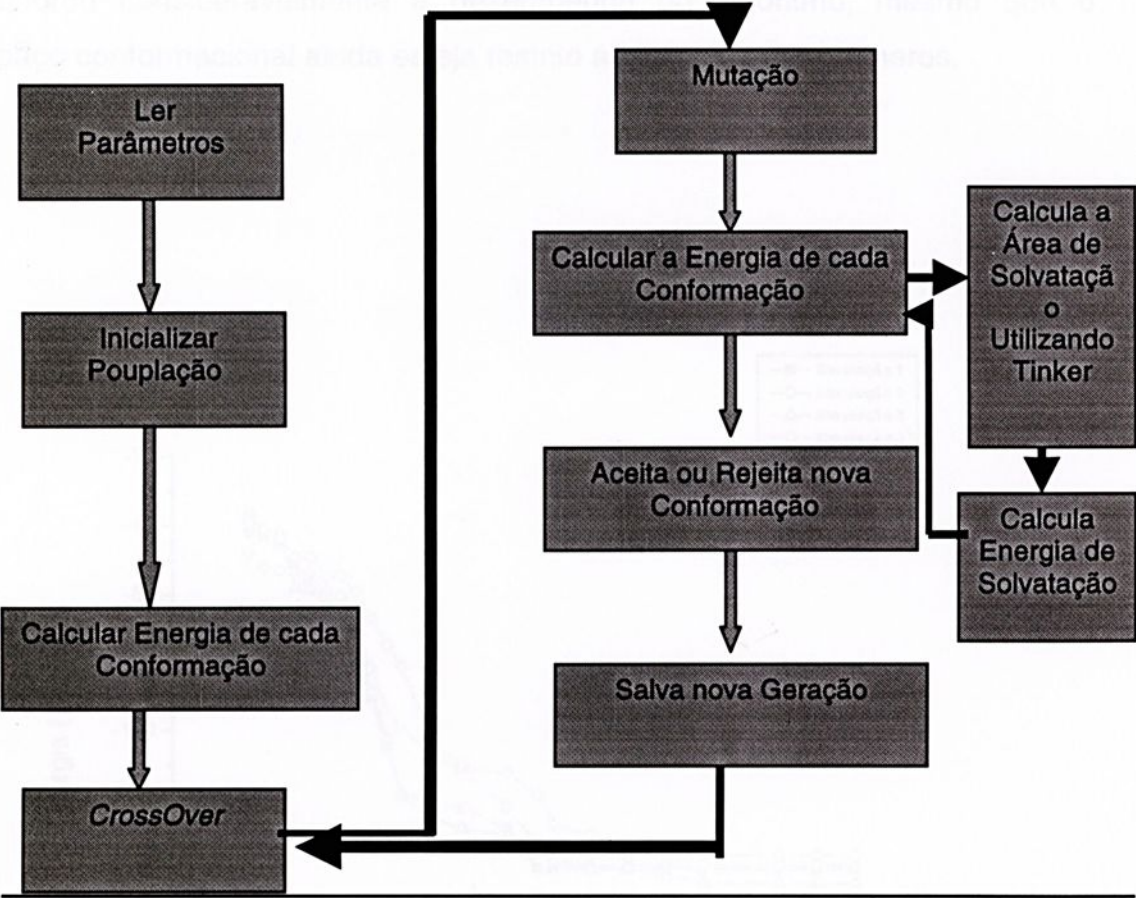


Figura 35- Esquema do Algoritmo Genético com Cálculo de Solvatação usando Tinker.

Utilizando o novo potencial, o algoritmo genético gerou conformações onde a cadeia lateral do Triptofano ajustou-se de forma bem melhor do que no caso

sem solvatação. Dessa maneira, uma melhor sobreposição das estruturas preditas pelo AG com a estrutura depositada no PDB.

Nas Figuras 36 e 37, são apresentados os gráficos, respectivamente, de energia mínima por geração e de energia média por gerações em função do número de gerações. Pode-se observar que, no caso da energia mínima, o algoritmo encontra um mínimo local entre a 30^a e 40^a geração e estabiliza-se após 45 gerações. Na Figura 38, pode-se observar a sobreposição de uma das estruturas preditas, com menor energia, com a estrutura depositada no PDB. Pode-se observar que houve uma melhor sobreposição do resíduo triptofano, o que indica que a inclusão de um termo de solvatação implícito melhorou consideravelmente o desempenho do algoritmo, mesmo que o espaço conformacional ainda esteja restrito a biblioteca de rotâmeros.

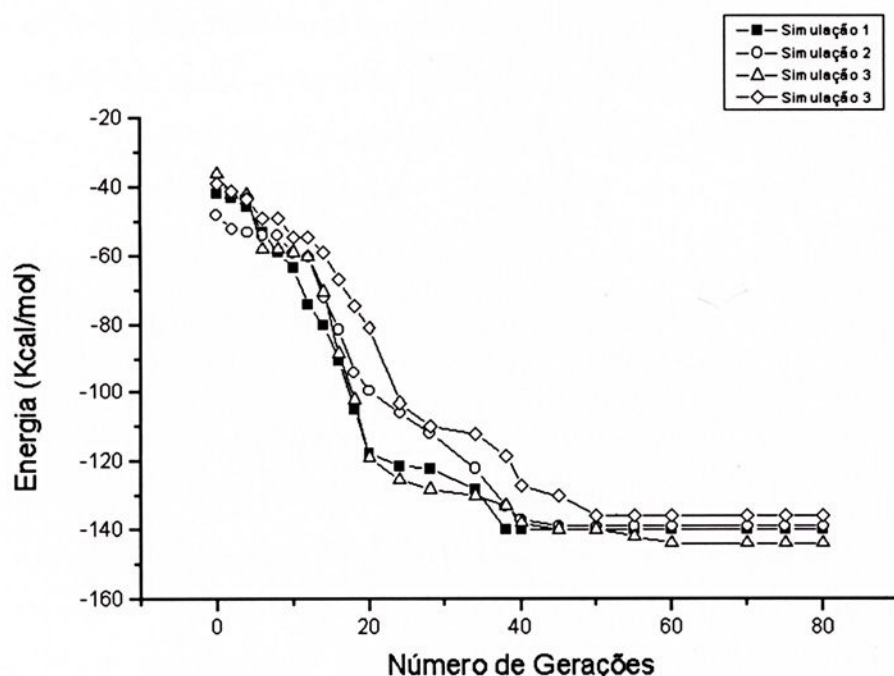


Figura 36- Gráfico, com o resultado de algumas das simulações realizadas, apresentando a Energia mínima por geração em função do número de gerações.

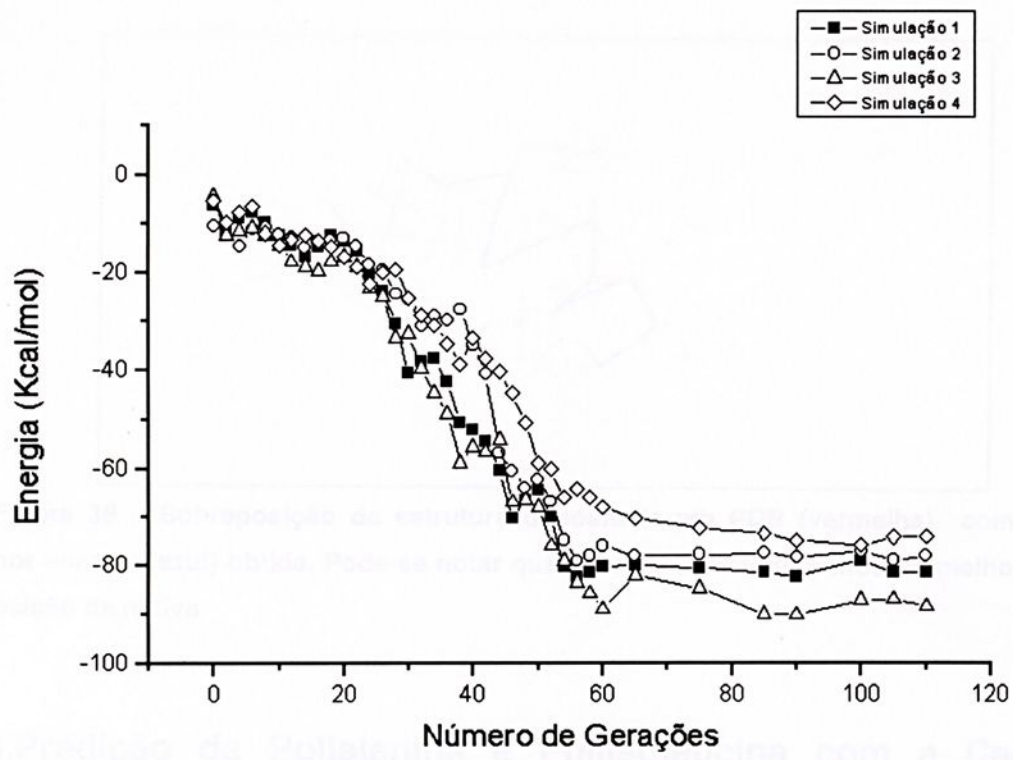


Figura 37- Gráfico, com o resultado de algumas das simulações realizadas, apresentando a Energia média por geração em função do número de gerações.

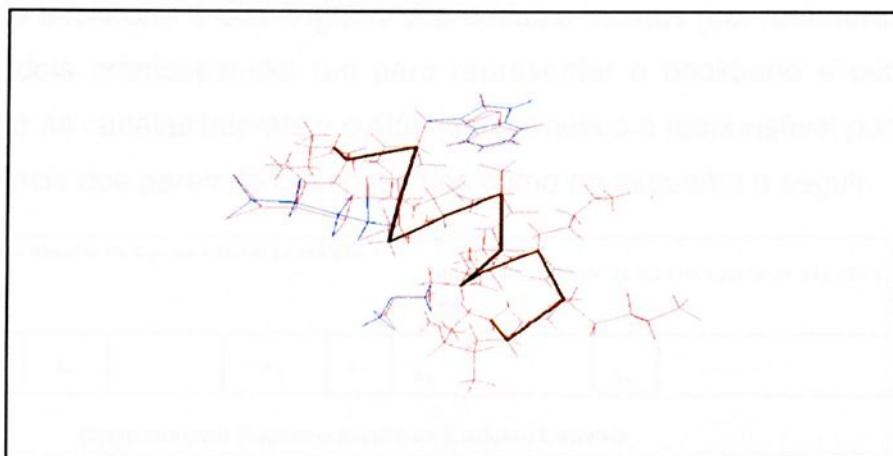


Figura 38 – Sobreposição da estrutura depositada em PDB (vermelha) com a de menor energia (azul) obtida. Pode-se notar que agora o triptofano concorda melhor com a posição da nativa

6.3.Predição da Polialanina e Poliisoleucina com a Cadeia Principal Relaxada

Considerando as limitações de ter a cadeia principal fixa, foi desenvolvida uma versão na qual a cadeia principal e as cadeias laterais eram ambas relaxadas. Uma Biblioteca de rotâmeros dependentes do backbone foi utilizada (Dunbrack & Karplus, 1994).

Nesse caso a biblioteca de rotâmeros fornece para cada aminoácido e para cada combinação de par de ângulos $[\phi, \psi]$ variando de 10 em 10 graus dentro do intervalo de $[-180, +180]$, os quatro valores dos ângulos χ_i mais prováveis. Essa biblioteca pode ser encontrada no endereço: <http://www.fccc.edu/research/labs/dunbrack/bbdep.html>

No algoritmo genético no qual a cadeia principal foi relaxada, cada indivíduo (cromossomo) representa uma conformação do peptídeo através do aminoácido e do rotâmero ativado para cada posição da cadeia com seus ângulos χ_i e a estrutura do backbone através dos ângulos diedrais $[\phi, \psi]$. Utilizando coordenadas internas, o algoritmo obtém as coordenadas cartesianas apenas para o cálculo da energia e para montar os arquivos PDBs

de saída. Todo o algoritmo genético trabalha em termos de coordenadas internas do backbone e dos ângulos das cadeias laterais (i.e. rotâmeros). São utilizados dois cromossomos: um para representar o backbone e outro para representar as cadeias laterais e o algoritmo genético é responsável por manter a consistência dos pares de cromossomos como no esquema a seguir.

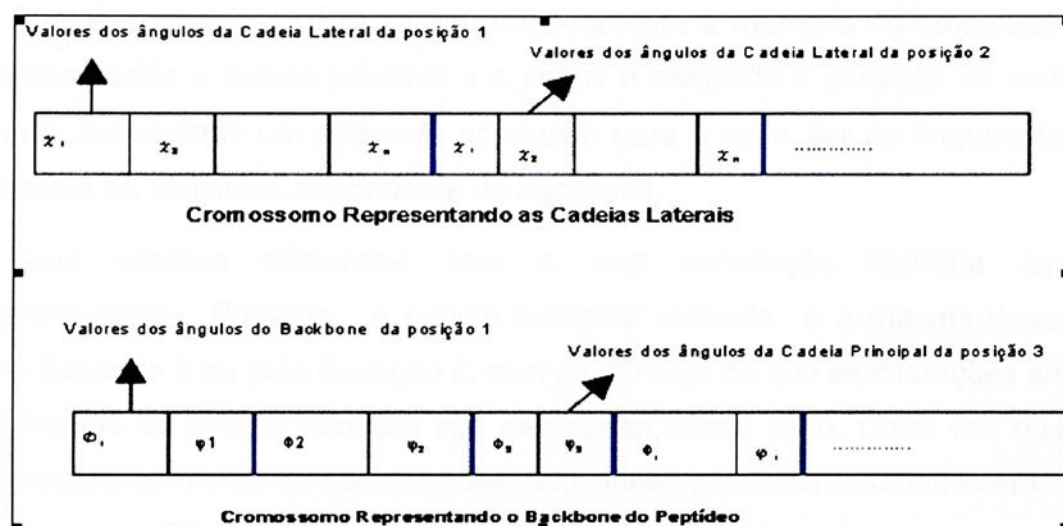


Figura 39 - Representação (codificação) das cadeias laterais e do backbone em pares de cromossomos.

Nesse algoritmo genético, diferentemente do programa para predição de *core*, as conformações não foram codificadas em binários nos cromossomos. Ou seja, foi utilizada uma forma adaptada de algoritmo genético, já que a codificação binária acarretaria em custo computacional, maior consumo de memória e nesse caso não auxiliaria muito na solução do problema. É importante ressaltar o fato de não ter codificado em binário não acarreta nenhuma perda de eficiência nos algoritmos genéticos nesse caso.

O núcleo do algoritmo genético para predição de estrutura terciária de peptídeos com o backbone e a cadeia lateral relaxados ao mesmo tempo é semelhante ao desenvolvido nos casos anteriores. Utiliza-se uma população contendo N indivíduos (conformações), sendo que a população inicial é gerada de maneira totalmente aleatória.

O cruzamento nesse caso é realizado entre os cromossomos que representam a cadeia principal da conformação sendo que o cromossomo que

representa a conformação das cadeias acompanha o mesmo cruzamento. De maneira que o cruzamento é realizado nas mesmas posições tanto para a representação da cadeia principal como da cadeia lateral. A posição do cruzamento é escolhida aleatoriamente e o critério de aceitação dos filhos é o mesmo dos algoritmos descritos anteriormente (aleatoriamente).

Quanto à mutação, é realizada primeiramente a mutação no cromossomo representando a cadeia principal e a seguir é realizada a mutação da cadeia lateral, escolhendo um rotâmetro adequado para o novo par de ângulos $[\phi, \phi]$ presente na biblioteca dependente do backbone.

Duas versões diferentes: com e sem solvatação implícita foram implementadas. Portanto, a função potencial utilizada é a mesma descrita pela Equação 3 ou pela Equação 2, com a diferença de que as interações entre os átomos da cadeia principal são calculadas nesse caso. Cada vez que é necessário converter de coordenadas cartesianas para coordenadas internas e vice-versa, o TINKER é chamado através do sistema operacional para auxiliar nessa conversão. Devido a isso, e ao fato de calcularmos interações entre todos os átomos do backbone, e ainda termos que representar e tratar tanto o backbone quanto as cadeias laterais no algoritmo genético, essa versão consome mais tempo de processamento do que quando relaxamos apenas as cadeias laterais.

Os arquivos de entrada desse software são diferentes da versão implementada com o backbone fixo e são os seguintes:

- **rotamer.dat** que contém as coordenadas internas do conjunto de rotâmetros dependente do backbone (Ponder & Richards, 1987), esse arquivo **rotamer.dat** é montado de forma automática através de um pequeno programa implementado em C que escolhe da base de rotâmetros, aqueles correspondentes aos aminoácidos presentes no peptídeo.
- **backbone.dat**: contém as coordenadas cartesianas do backbone extraídas, automaticamente, do arquivo PDB fornecido pelo usuário. Esse arquivo é gerado automaticamente sem intervenção do usuário

e é opcional. Caso o arquivo PDB não seja fornecido ou não exista, um arquivo é construído a partir da sequência utilizando a função BUILD do TINKER e otimizado e a seguir, o backbone.dat é extraído.

- o wal.dat: contém os parâmetros de van der waals para cada tipo de átomo do PDB e a carga parcial de cada átomo também.
- Param.dat: permite o usuário parametrizar sua simulação. Entre outras coisas, através desse arquivo é possível: (i) definir o tamanho da população, a porcentagem de cromossomos que são copiados diretamente de uma geração para outra; (ii) definir a taxa de mutação; (iii) definir o número máximo de gerações; (iv) especificar o número de rotâmeros máximo por aminoácido; (v) definir o número de resíduos do peptídeo.

Esse algoritmo genético foi utilizado para prever a estrutura de dois peptídeos: polialanina e poliisoleucina. Num primeiro momento, testamos a polialanina composta de 20 monômeros. Pode-se observar pelos gráficos, das Figuras 40 e 41, que a energia mínima estabiliza-se após 700 gerações aproximadamente, no caso do algoritmo sem solvatação e após 600 gerações no caso com solvatação. Pode-se observar ainda que ambos algoritmos genéticos (com e sem solvatação) encontraram alguns mínimos locais.

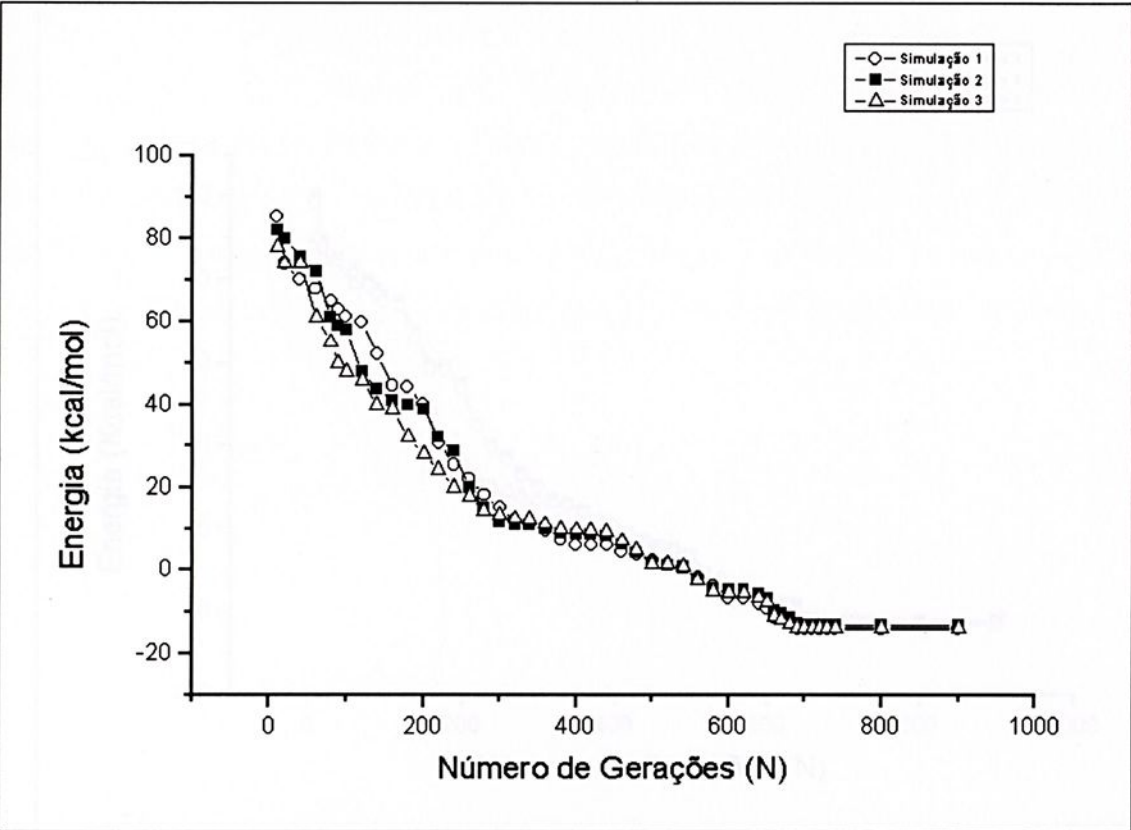


Figura 40- Energia Mínima da Simulação da Polialanina com um termo de Energia de Solvatação e o Backbone Relaxado.

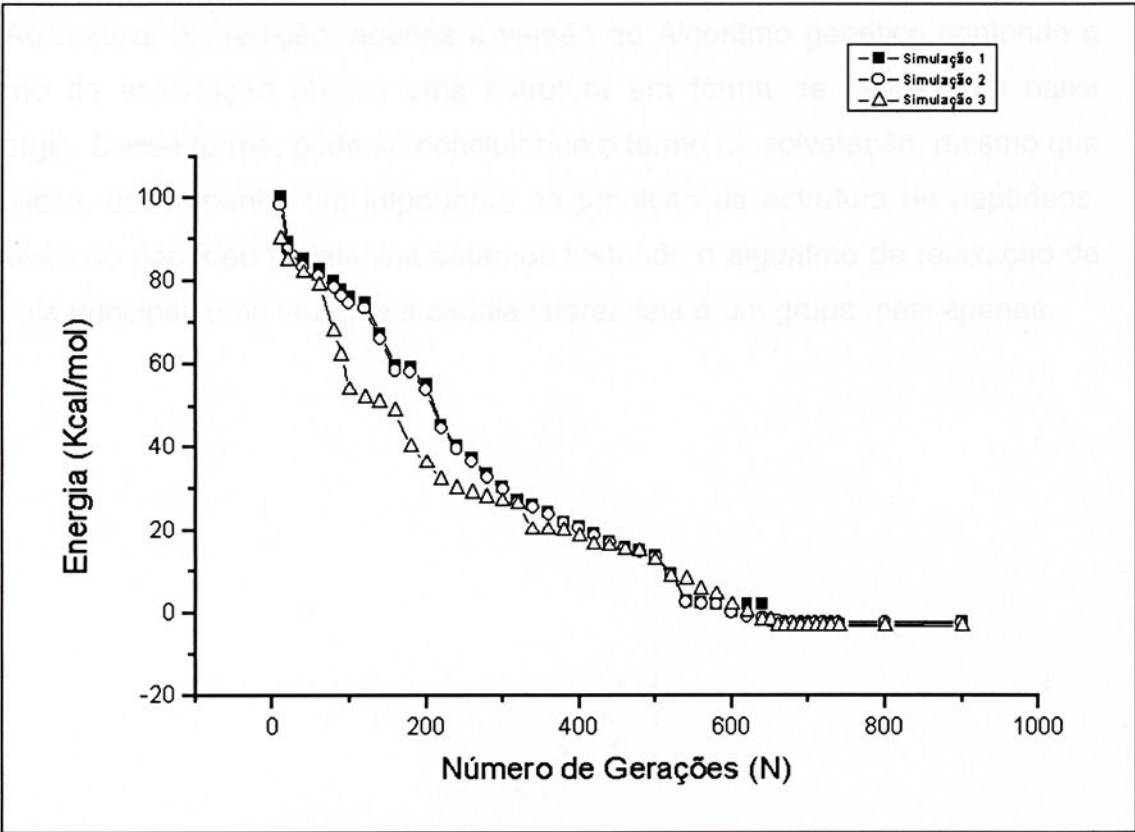


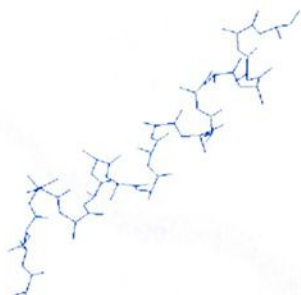
Figura 41- Energia Mínima da Simulação da Polialanina sem o termo de Energia de Solvatação e o Backbone Relaxado.

Figura 42- Estrutura de helix obtida pela predição de uma polialanina utilizando-se o AG.

Na figura 42, é apresentada uma das estruturas de menor energia, por volta de -14 kcal/mol, obtida pelo algoritmo genético com o backbone relaxado. Nesse caso, a energia é negativa, o que indica que a estrutura é estável. A energia de solvatação é considerada zero, pois não há interação com o meio aquoso. O valor de -14 kcal/mol é muito baixo, o que indica que a estrutura é muito estável. Isso pode ser devido ao fato de que a polialanina é uma proteína muito simples, com apenas 20 aminoácidos, e a estrutura de helix é a mais comum e estável.

Em um segundo passo, foi utilizado o mesmo algoritmo, porém apenas a variação de solvatação, para prever a estrutura de uma polialanina composta de 18 resíduos. Nesse caso, o algoritmo variou um espaço de conformações muito maior, devido ao fato de que há mais combinações entre os pares de aminoácidos (aa) e o backbone. A Figura 43 apresenta o gráfico de

Ao realizar a predição, apenas a versão do Algoritmo genético contendo o termo de solvatação obteve uma estrutura em forma de hélice com baixa energia. Dessa forma, pode-se concluir que o termo de solvatação, mesmo que implícito, desempenha um importante na predição da estrutura de peptídeos. Através do peptídeo polialanina estamos testando o algoritmo de relaxação da cadeia principal, uma vez que a cadeia lateral dela é um grupo metil apenas.



**Figura 42- Estrutura de hélice obtida pela predição de uma polialanina
Utilizando um AG.**

Na figura 42, é ilustrada uma das estruturas de mais baixa energia, por volta de -14 Kcal/mol, obtida pelo algoritmo genético com o backbone relaxado, biblioteca de rotâmeros dependente da cadeia principal e com um termo de solvatação implícito. Deve-se notar que essa polialanina é composta de 20 resíduos dos quais aproximadamente 16 estão em hélice.

Em um segundo passo foi utilizado o mesmo algoritmo, porém apenas a versão com solvatação, para prever a estrutura de uma poliisoleucina composta de 18 resíduos. Nesse caso o algoritmo vasculha um espaço de conformações bastante vasto devido as possíveis combinações entre os pares de ângulos $[\varphi, \psi]$ e o possíveis ângulos χ_i . A Figura 43 apresenta o gráfico de

energia mínima por geração, também é possível observar a presença de mínimos e que a energia estabiliza-se após 700 gerações, permanecendo estável. Pode-se observar ainda, no gráfico, que diferentes simulações possuem um comportamento semelhante a atinge patamares de Energia mínima muito próximos.

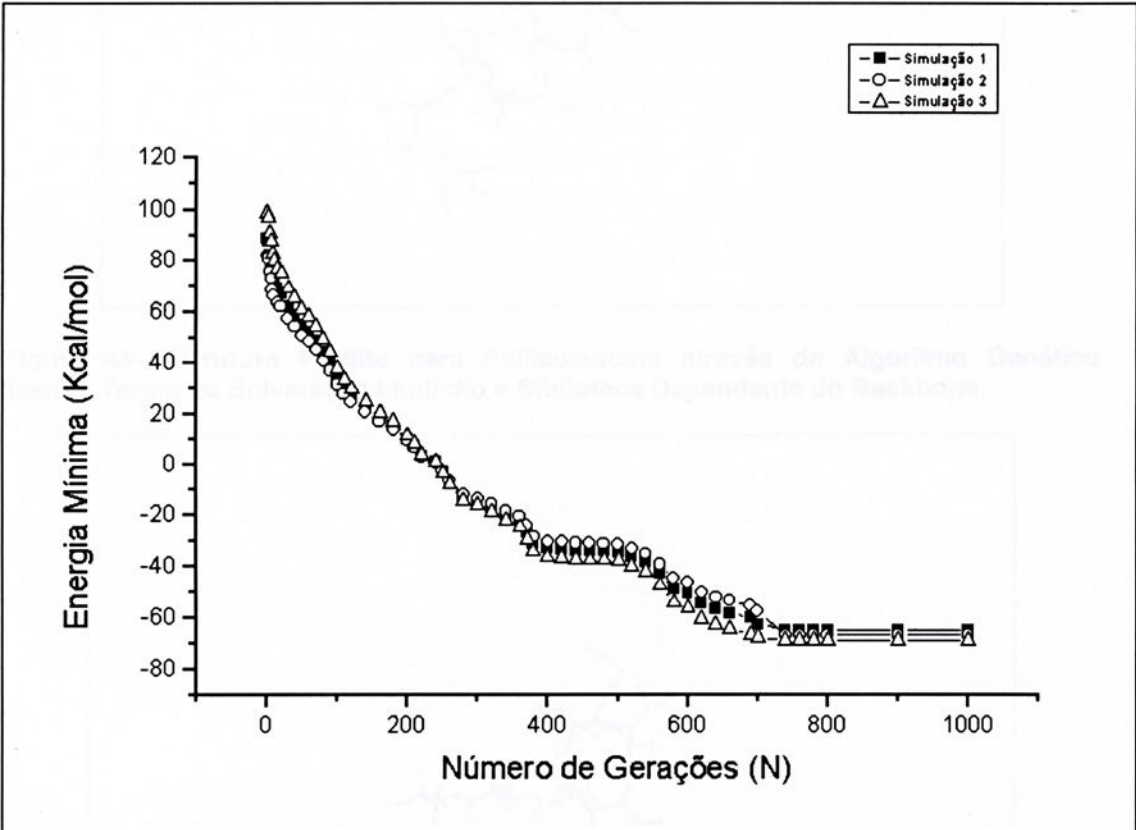


Figura 43- Energia Mínima da Simulação da Poliisoleucina com o termo de Energia de Solvatação e o Backbone Relaxado.

Nas Figuras 44 e 45 estão apresentadas duas estruturas de baixa energia que foram preditas pelo algoritmo (Figura 45 e 46), ambas apresentam um trecho formando a estrutura secundária de hélice.

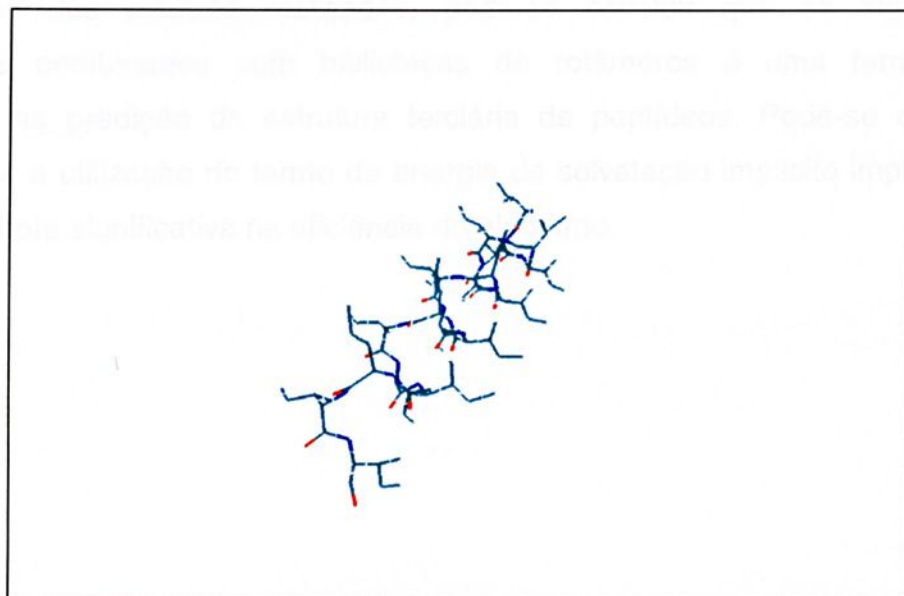


Figura 44- Estrutura Predita para Poliisoleucina através de Algoritmo Genético Utilizando Termo de Solvatação Implícito e Biblioteca Dependente do Backbone.

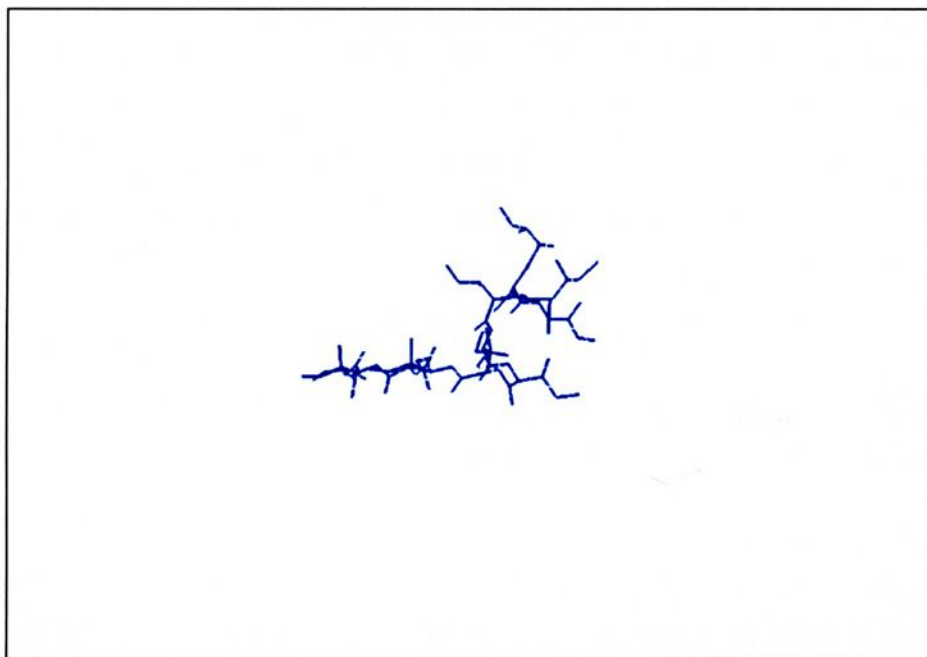


Figura 45- Estrutura Predita para Poliisoleucina através de Algoritmo Genético Utilizando Termo de Solvatação Implícito e Biblioteca Dependente do Backbone.

Uma das estruturas preditas apresenta um α -hélice, enquanto a outra apresenta um pedaço em α -hélice e uma dobra sobre si mesma. Uma pequena dinâmica molecular sobre essa segunda estrutura foi realizada, porém ela manteve-se praticamente inalterada.

Através dos estudos realizados, pode-se concluir que os algoritmos genéticos combinados com bibliotecas de rotâmeros é uma ferramenta eficiente na predição da estrutura terciária de peptídeos. Pode-se concluir ainda que a utilização do termo de energia de solvatação implícito implica em uma melhora significativa na eficiência do algoritmo.

Predição 1D de Proteínas Usando Redes Neurais

1.1 Conceitos Básicos

Uma rede neural é uma coleção de neurônios artificiais que se conectam entre si, formando uma estrutura que pode ser usada para modelar a relação entre uma entrada e uma saída. A rede neural é composta por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada neurônio em uma camada está conectado a todos os neurônios na camada seguinte. A rede neural é treinada com um conjunto de dados de entrada e saída, e o objetivo é encontrar os pesos das conexões entre os neurônios que minimizem a diferença entre a saída prevista e a saída real.

Uma rede neural pode ser usada para modelar a relação entre uma entrada e uma saída, e pode ser treinada com um conjunto de dados de entrada e saída. A rede neural é composta por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada neurônio em uma camada está conectado a todos os neurônios na camada seguinte. A rede neural é treinada com um conjunto de dados de entrada e saída, e o objetivo é encontrar os pesos das conexões entre os neurônios que minimizem a diferença entre a saída prevista e a saída real.

Uma rede neural pode ser usada para modelar a relação entre uma entrada e uma saída, e pode ser treinada com um conjunto de dados de entrada e saída. A rede neural é composta por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada neurônio em uma camada está conectado a todos os neurônios na camada seguinte. A rede neural é treinada com um conjunto de dados de entrada e saída, e o objetivo é encontrar os pesos das conexões entre os neurônios que minimizem a diferença entre a saída prevista e a saída real.

Uma rede neural pode ser usada para modelar a relação entre uma entrada e uma saída, e pode ser treinada com um conjunto de dados de entrada e saída. A rede neural é composta por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada neurônio em uma camada está conectado a todos os neurônios na camada seguinte. A rede neural é treinada com um conjunto de dados de entrada e saída, e o objetivo é encontrar os pesos das conexões entre os neurônios que minimizem a diferença entre a saída prevista e a saída real.

Capítulo 7

Predição 1D de Proteínas Usando Redes Neurais

7.1.Considerações Iniciais

Nesse capítulo é discutida a aplicação de redes neurais do tipo Multi Layer Perceptron na predição de estruturas secundárias de proteínas. Redes do tipo MLP foram aplicadas para predizer a estrutura secundária fornecida à seqüência de aminoácidos e propriedades físico-químicas. Pode-se notar que a arquitetura da rede neural, o banco de dados utilizado e a codificação da entrada das redes neurais são extremamente importantes para a performance da predição. Desenvolveu-se um sistema de predição 1D.

Para o desenvolvimento de um modelo ou projeto baseado em redes neurais artificiais, cujo problema está definido, foram necessárias diversas etapas (Figura 46):

- Coleta dos dados de treinamento, validação e teste: devem ser reunidos todos os dados pertinentes e potencialmente úteis à tarefa. Os dados coletados devem ser separados em conjuntos de treinamento, validação e teste, para serem utilizados em cada uma das fases do desenvolvimento da rede.
- Pré e Pós-processamento dos dados: os dados simbólicos devem ser transformados em dados puramente numéricos, os quais são mais adequados para utilização pela rede.

- Projeto da estrutura da rede: a escolha da configuração adequada da rede tem um impacto substancial no desempenho do sistema.
- Treinamento, teste e validação: o treinamento é realizado em diferentes arquiteturas de rede, e para estas arquiteturas se realiza a validação do treinamento e os testes para avaliar o melhor desempenho.

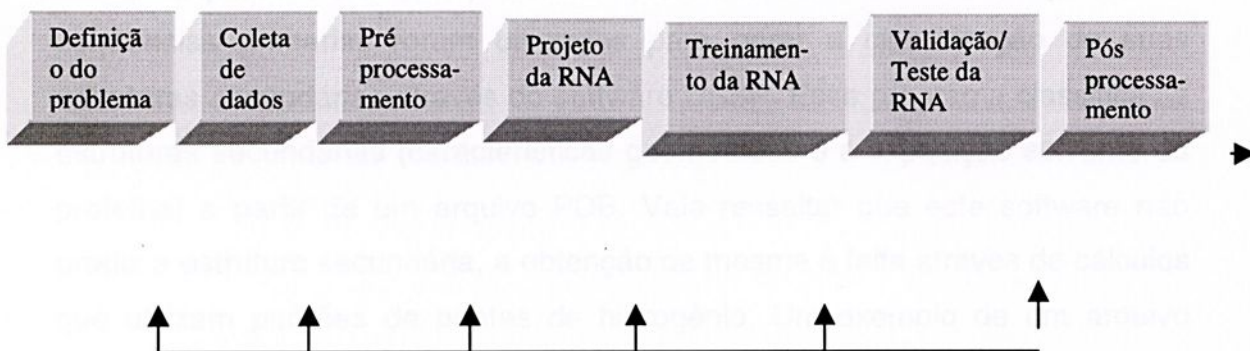


Figura 46 –Desenvolvimento de um projeto baseado em RNA.

7.2.Bases de Dados

O primeiro passo para a implementação do projeto foi à coleta e seleção de três conjuntos de seqüências primárias de proteínas. A todos os conjuntos de proteínas foram aplicados o alinhamento múltiplo para evitar a presença de proteínas homólogas. Para o conjunto de treinamento foram selecionadas 389 proteínas. Estas proteínas foram divididas em 4 subconjuntos (todas, hélice, folha e hélice-folha) para o treinamento de RNAs específicas.

O segundo conjunto de proteínas foi obtido através do EVA (Evaluation of Automatic protein structure prediction)¹. Este órgão possui um servidor de predição de estruturas de proteínas em tempo real que prediz as seqüências depositadas no PDB toda semana, e faz comparações com outros servidores². Em particular foram utilizadas para o segundo conjunto as proteínas publicadas em 27/02/2002.

¹Site: <http://cubic.bioc.columbia.edu/eva/doc/concept.html>

²Existem outros servidores de predição de estruturas como o CASP, PREDICTOR e DSPRED
<http://www.bioinfo.rpi.edu/~bystre/hmmstr/server.php>
<http://predictioncenter.llnl.gov>.

O terceiro conjunto de proteínas foi obtido através do artigo publicado por Ning Qian e Terrence J. Sejnowski (Quian & Sejnowski, 1988). Este conjunto totaliza 75 proteínas não idênticas que serão utilizadas para testar as diferentes arquiteturas de RNA e obter o desempenho das mesmas através de coeficientes conhecidos na literatura, com efeito de comparação.

Os arquivos PDB das proteínas que constituem os três conjuntos de seqüências primárias foram utilizados para gerar a classificação de suas estruturas secundárias através do software DSSP. Esse programa classifica as estruturas secundárias (características geométricas e a exposição solvente da proteína) a partir de um arquivo PDB. Vale ressaltar que este software não prediz a estrutura secundária, a obtenção da mesma é feita através de cálculos que utilizam padrões de pontes de hidrogênio. Um exemplo de um arquivo DSSP pode ser visto no Anexo.

O programa DSSP classifica cada resíduo em oito classes (H = *alpha helix*; B = *residue in isolated β -bridge*; E = *extended strand*; G = *3-helix [3/10 helix]*; I = *5 helix [*pi* helix]*; T = *hydrogen bonded turn*; S = *bend*; e "." = indefinido) (Figura 47). Estas classes são tipicamente agrupadas em três classes padronizadas que são: hélices, β -folha e *coil*. A classe hélices contém H, G e I, a classe β -folha contém E e B, e a classe *coils* contém todas as outras.

8 classes	H	G	I	E	B	S	T	"."
3 classes	H	H	H	E	E	C	C	C

Figura 47- Mapeamento Clássico das oito classes do DSSP para 3 classes.

Com a codificação do problema e a coleta dos dados já definidos, implementou-se um software de interface que realiza a conversão de arquivos

PDB em arquivos que contenham somente as três classes de estruturas secundárias e que realiza o pré-processamento para projetar as RNAs.

7.3. Interface do Pré-Processamento

Para realizar a interface foi implementado, por nós, em C++ Builder 5, um software chamado de Conversor (v. figura 48). Ele realiza as seguintes tarefas:

- Conversão de vários arquivos PDB para arquivos DSSP automaticamente, mas deve-se ter o programa dsspcmbi.exe³;
- Através dos arquivos DSSP, obtêm-se as três classes que serão utilizadas, e também se divide bases de proteínas de acordo com sua porcentagem de estruturas, ou seja, em: todas, hélice, folha e hélice-folha. Estas bases serão muito importantes para os treinamentos a serem realizados;
- Criação de arquivos que descrevem as janelas de proteínas que serão inseridas nas redes neurais;
- Geração de arquivos de estatísticas sobre as proteínas, que são úteis para verificar as características das bases criadas;
- Geração das matrizes de entrada e saída para a utilização das mesmas nas RNAs.

O arquivo dsspcmbi.exe é o software DSSP que é executado somente em ambiente DOS-Windows e com um arquivo de entrada PDB.



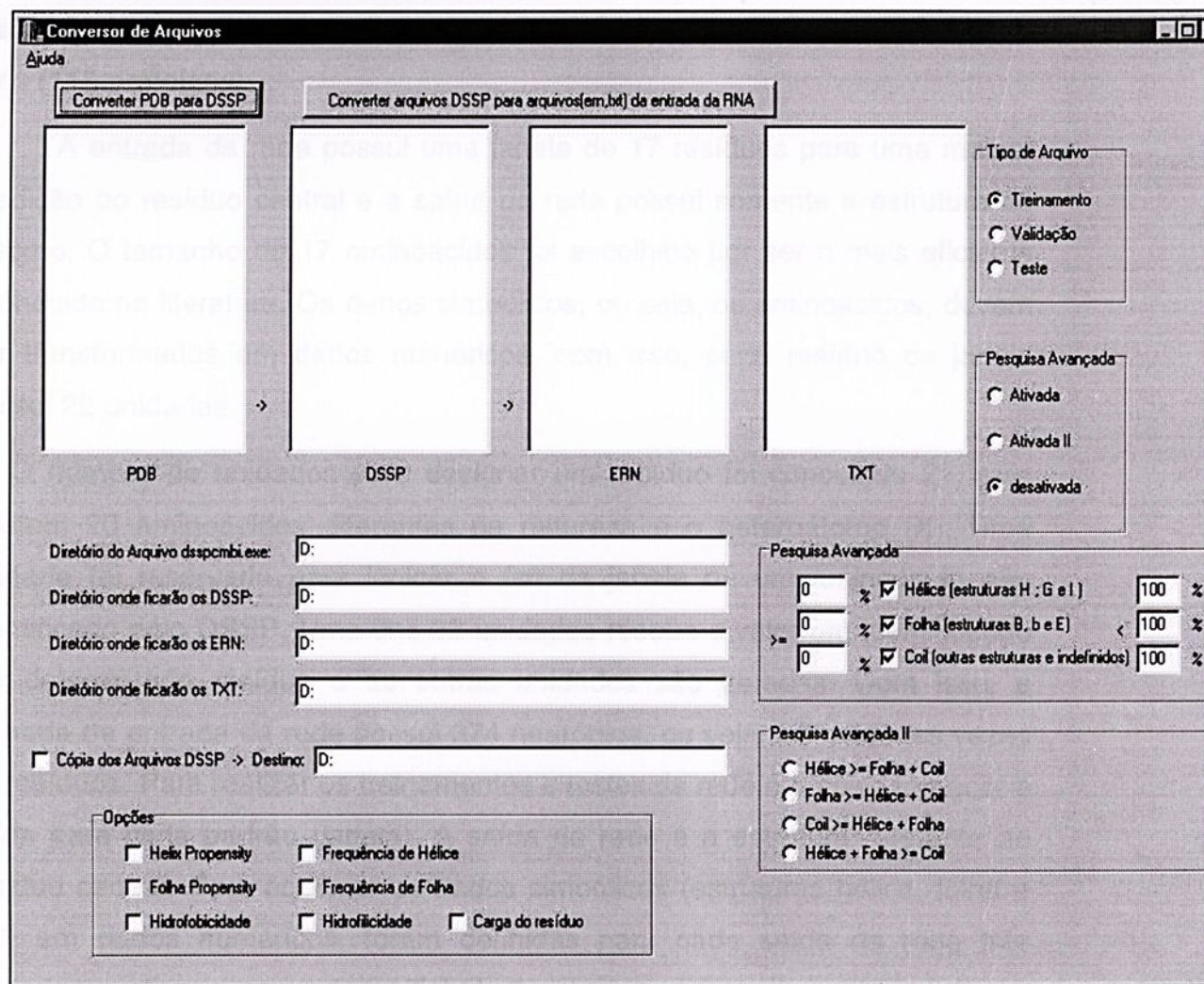


Figura 48 - Software de Interface Conversor.

O conjunto de treinamento foi dividido em subconjuntos, através do software Conversor, para a realização de testes específicos. Os subconjuntos são: **Todas** (389 proteínas), o qual possui todas as proteínas coletadas; **Hélice**, que contém proteínas cujo número de resíduos em estrutura hélice é maior que a soma de resíduos em estrutura folha e resíduos em estrutura *coil* (173 proteínas); **Folha**, que possui proteínas cujo número de resíduos em estrutura folha é maior que a soma de resíduos em estrutura hélice e resíduos em estrutura *coil* (56 proteínas); e **Hélice-Folha**, ou seja, proteínas cuja porcentagem de resíduos em estrutura hélice é superior a 30% e inferior a 50%

e a porcentagem de resíduos em estrutura folha é superior a 30% e inferior a 50% (115 proteínas).

A entrada da rede possui uma janela de 17 resíduos para uma melhor predição do resíduo central e a saída da rede possui somente a estrutura do mesmo. O tamanho de 17 aminoácidos foi escolhido por ser o mais eficiente conhecido na literatura. Os dados simbólicos, ou seja, os aminoácidos, devem ser transformados em dados numéricos, com isso, cada resíduo da janela possui 22 unidades.

O número de unidades para designar um resíduo foi concebido 22, pois existem 20 aminoácidos diferentes na natureza e o heteroátomo (X). Uma unidade foi reservada para indicar o fim da janela ou um aminoácido não identificado pelo DSSP. Uma das 22 unidades recebe o valor um identificando um determinado resíduo e as outras unidades são zeradas. Com isso, a camada de entrada da rede possui 374 neurônios, ou seja, 22 unidades vezes 17 resíduos. Para realizar os treinamentos e testes da rede é preciso fornecer a saída para cada padrão (janela). A saída da rede é a estrutura referente ao resíduo central. Para codificar os dados simbólicos (estruturas hélice, folha e coil) em dados numéricos, foram definidas para cada saída da rede três unidades conforme mostrado na Tabela 5.

Estrutura	Codificação
Hélice	1 0 0
Folha	0 0 1
Coil	0 1 0

Tabela 5 – Codificação Binária das Estruturas Secundárias.

Através destas informações, o software cria os arquivos de entrada e saída das RNAs. Estes arquivos são matrizes em que as linhas representam os padrões, as colunas da entrada são os números codificados dos resíduos da janela e as colunas da saída são a codificação da estrutura do resíduo central da janela.

7.4. Coeficientes

Existem alguns coeficientes para verificar o desempenho de uma RNA na predição de estruturas secundárias. Dentre estes, o mais utilizado pelos pesquisadores é conhecido como Q3, que fornece a porcentagem dos resíduos preditos corretamente considerando os três tipos de estruturas secundárias: hélice, folha e coil (Chandonia & Karplus, 1999).

$$\text{Coeficiente Q3 é dado por : } Q3 = \frac{\sum_{(i=H,E,C)} CORRi}{\sum_{(i=H,E,C)} TOTi} * 100, \text{ onde } i = \{\text{hélice, folha,}$$

coil\}, TOTi é o número de "i" existentes nas proteínas de teste, PREDi é número total

de "i" que a RNA predisse e CORRi = número de "i" que a RNA predito corretamente.

Outros coeficientes importantes e também utilizados no projeto para ajudar na avaliação dos testes são o Q[obs] e Q[pred]. O primeiro dá a porcentagem do número de resíduos preditos corretamente em relação ao número real observado, em um estado particular. Já o Q[prd] dá a porcentagem do número de resíduos preditos corretamente em relação ao número que a RNA predisse em um estado particular. Onde $Qi[obs] = \frac{CORRi}{TOTi} * 100$ e

$$Qi[prd] = \frac{CORRi}{PREDi} * 100.$$

Para a implementação destes coeficientes que avaliam o desempenho da rede, foi desenvolvido, por nós, um software, em C++ Builder 5, chamado de Comparar. Este software fornece informações como porcentagem de acerto,

linhas incompatíveis, estruturas preditas e coeficientes (Figura 49). Esse software é responsável pelo pós-processamento.

Comparar estruturas

Passar de números para estrutura:

Abrir -> Estrutura

Saída:

Comparar saída da RNA com a estrutura real:

Abrir Est. Real

Abrir Saída RNA

Comparar

Total de linhas:

Linhas Incompatíveis:

Porcentagem de Acerto (Q3): %

Nro Helice:

Pred Helice:

Qh (obs): %

Qh (prd): %

Nro Folha:

Pred Folha:

Qf (obs): %

Qf (prd): %

Nro Coit:

Pred Coit:

Qc (obs): %

Qc (prd): %

Figura 49 - Software Comparar.

7.5. Simulação Utilizando uma única Rede Neural Artificial

Todas RNAs foram projetadas e treinadas no Simulador MATLAB 5.0. Este software é bastante conceituado pela comunidade científica e possui várias ferramentas para implementações de redes neurais (*Neural Network Toolbox*). As RNAs implementadas foram do tipo *Multi Layer Perceptron* (MLP) com a função de treinamento *trainrp*, que utiliza o algoritmo *backpropagation* (RPROP).

Na elaboração de um projeto de RNA ideal para solucionar um determinado problema, uma das tarefas mais difíceis é determinar o número de neurônios de processamento da camada intermediária (oculta), bem como o número de camadas intermediárias. Não existem regras para isto, entretanto, estudos

demonstraram que uma única camada intermediária é suficiente para representar qualquer função ou para resolver a maioria dos problemas de generalização.

Neste projeto, foram utilizados apenas 2 ou 3 camadas intermediárias, variando o número de neurônios utilizados em cada uma destas camadas (v. Figura 50). Em todas as redes, os neurônios de entrada, ou seja, a camada de entrada utilizou a função de transferência linear (*purelim*), já as demais camadas utilizaram as funções de transferência tangente hiperbólica sigmoidal (*tansig*) e logarítmica sigmoidal (*logsig*). As primeiras simulações foram realizadas com uma arquitetura composta por apenas uma rede neural descrita na Figura 50.

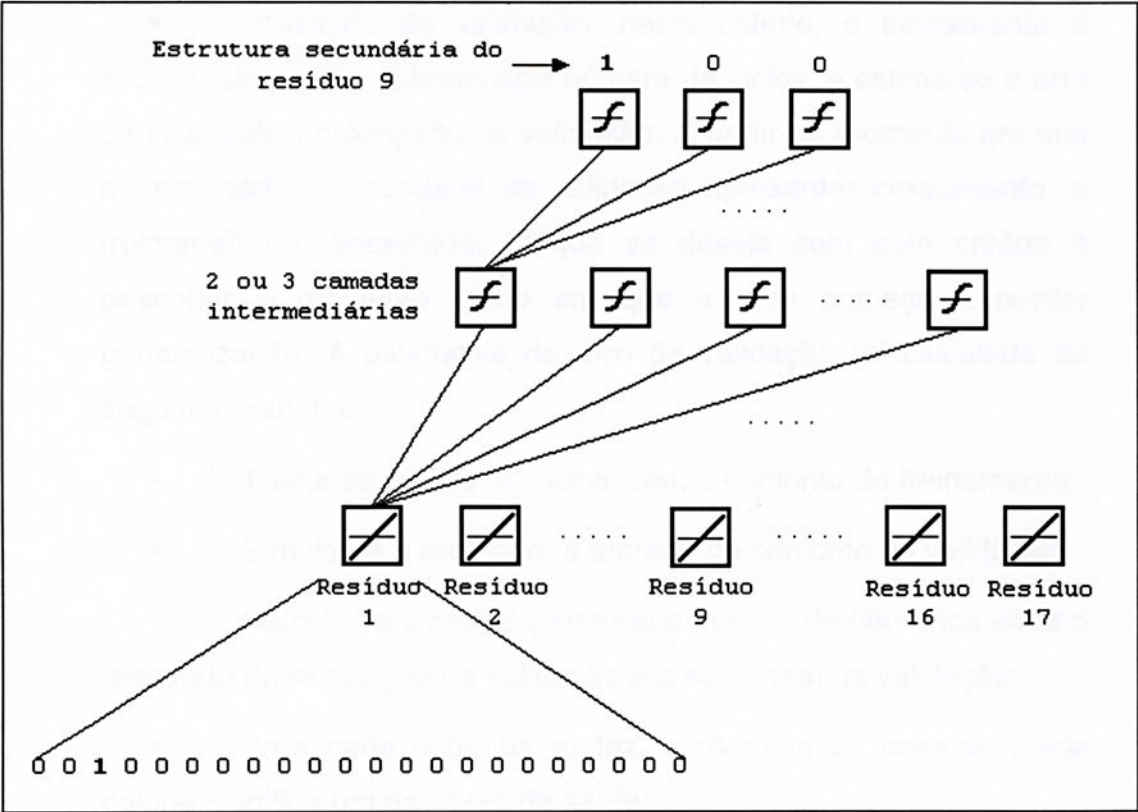


Figura 50 - Arquitetura com uma Rede Neural Artificial.

No processo de treinamento, os elementos de processamento (neurônios das camadas) da RNA são inicializados aleatoriamente com diferentes limiares

internos e, as conexões entre eles com pequenos pesos (valores que variam entre $-1.0 < w < 1.0$, onde w denota o peso de uma conexão qualquer). Em seguida, os dados de treinamento, ou seja, todos os padrões existentes na entrada e na saída do conjunto de treinamento, são apresentados à RNA repetidamente e, a cada ciclo de treinamento os pesos são ajustados.

Existem vários métodos para a determinação do momento em que o treinamento de uma rede neural deve ser encerrado. Uma boa determinação destes critérios é fundamental para um bom treinamento e conseqüentemente uma boa generalização. Os critérios de parada utilizados no projeto, em ordem de prioridade, são:

- O treinamento é interrompido quando o erro quadrático médio do treinamento for menor que 0.01 (1% de erro);
- Utilização da validação: neste critério, o treinamento é interrompido a um determinado número de ciclos, e estima-se o erro da rede sobre o conjunto de validação. A partir do momento em que o erro médio no conjunto de validação apresentar crescimento, o treinamento é encerrado. O que se deseja com este critério é descobrir o momento exato em que a rede começa a perder generalização. A estimativa de erro da validação foi calculada da seguinte maneira:
 - Treina-se a rede “x” ciclos, com o conjunto de treinamento;
 - Simula-se a rede com a entrada do conjunto de validação;
 - Calcula-se a matriz contendo o módulo da diferença entre o resultado da simulação da validação e a saída real da validação;
 - Para cada linha da matriz, somam-se as colunas (cada coluna significa um neurônio de saída);
 - Cada linha que contiver a soma superior a 0.5, considerar como um erro de padrão da rede;
 - Soma-se o número de linhas do conjunto de validação, que tiveram erro de padrão;



- Se o número de linhas com erro no conjunto de validação, no passo atual, for maior que o número calculado anteriormente parar treinamento. Senão treina-se x ciclos novamente.

7.5.1 Resultados

A seguir são apresentados os resultados das predições realizada com redes com uma única camada e com arquiteturas e bases de treinamento distintas. As tabelas apresentam a base de treinamento utilizada, a rede utilizada (com suas funções de ativação), o erro de treinamento e o coeficiente Q3.

Base de Treinamento: Todas (216 proteínas). 4 matrizes de 54 proteínas.
Arquitetura (camadas): 4 camadas (374 – purelim; 24 - tansig; 12 - tansig; 3 – logsig)
Ciclos: 800 Erro de Treinamento: 9,8%
Porcentagem de Acerto (Q3): 50,6%
Hélice: Q[obs]: 37,0% Q[prd]: 51,0% ; Folha: Q[obs]: 36,0% Q[prd]: 45,0% ; Coil: Q[obs]: 64,0% Q[prd]: 63,0%

Base de Treinamento: Todas (216 proteínas). 4 matrizes de 54 proteínas.
Arquitetura (camadas): 4 camadas (374 – purelim; 24 - tansig; 8 - tansig; 3 – logsig)
Ciclos: 600 Erro de Treinamento: 10,2%
Porcentagem de Acerto (Q3): 46,0%
Hélice: Q[obs]: 35,0% Q[prd]: 56,0% ; Folha: Q[obs]: 40,0% Q[prd]: 47,0% ; Coil: Q[obs]: 54,0% Q[prd]: 67,0%

Base de Treinamento: Hélice (173 proteínas). 2 matrizes de 54 proteínas e uma de 65.
Arquitetura (camadas): 4 camadas (374 – purelim; 24 - tansig; 8 - tansig; 3 – tansig)
Ciclos: 400 Erro de Treinamento: 7,5%
Porcentagem de Acerto (Q3): 50,4%
Hélice: Q[obs]: 61,0% Q[prd]: 43,0% ; Folha: Q[obs]: 16,0% Q[prd]: 55,0% ; Coil: Q[obs]: 59,0% Q[prd]: 63,0%

Base de Treinamento: Hélice (173 proteínas). 2 matrizes de 54 proteínas e uma de 65.
Arquitetura (camadas): 4 camadas (374 – purelim; 24 - tansig; 12 - tansig; 3 – tansig)
Ciclos: 1000 Erro de Treinamento: 7,5%
Porcentagem de Acerto (Q3): 50,9%
Hélice: Q[obs]: 62,0% Q[prd]: 42,0% ; Folha: Q[obs]: 19,0% Q[prd]: 50,0% ; Coil: Q[obs]: 58,0% Q[prd]: 64,0%

Base de Treinamento: Folha (56 proteínas)
Arquitetura (camadas): 4 camadas (374 – purelim; 24 - tansig; 12 - tansig; 3 – tansig)
Ciclos: 1200 Erro de Treinamento: 1,6%
Porcentagem de Acerto (Q3): 47,4%
Hélice: Q[obs]: 8,0% Q[prd]: 58,0% ; Folha: Q[obs]: 70,0% Q[prd]: 35,0% ; Coil: Q[obs]: 58,0% Q[prd]: 65,0%

Base de Treinamento: Hélice-Folha (173 proteínas)
Arquitetura (camadas): 4 camadas (374 – purelim; 24 - tansig; 8 - tansig; 3 – tansig)
Ciclos: 800 Erro de Treinamento: 6,2%
Porcentagem de Acerto (Q3): 52,8%
Hélice: Q[obs]: 44,0% Q[prd]: 50,0% ; Folha: Q[obs]: 39,0% Q[prd]: 45,0% ; Coil: Q[obs]: 63,0% Q[prd]: 64,0%

Nestes experimentos, comprovou-se que nem sempre aumentando o número de camadas intermediárias ou de neurônios, melhora-se os resultados da rede. Os resultados reforçam a importância de uma arquitetura e uma base de dados bem projetada para obter uma boa predição 1D. Pode-se observar também que ao aumentar um pouco o número de ciclos o erro de treinamento diminui e o Q[prd] para hélice e folha chega a aumentar em 2%.

Pode-se observar que redes treinadas com diferentes bases de dados (toda- α , toda- β , todas e α/β) possuem performances diferentes para o mesmo conjunto de treinamento, o que já era de se esperar. Na tentativa de melhorar a performance da rede neural, mantendo a mesma base de dados implementou-se uma arquitetura com duas redes. Em termos de tempo, as redes são custosas para treinar devido ao fato de utilizar muitos padrões de entrada de treinamento e o MATLAB.

7.6. Implementação da Arquitetura com Duas Redes

Com o intuito de melhorar os resultados dos testes realizados com a arquitetura composta por apenas uma RNA, foi implementada uma nova arquitetura utilizando duas redes, uma seguida da outra. A primeira rede desempenha o papel descrito anteriormente, e a segunda rede combina as janelas de aminoácidos com os resultados da primeira rede (v. figura 51).

A segunda rede neural adiciona a informação de saída da primeira rede na sua entrada. Então a camada de entrada da segunda rede contém 425 neurônios, ou seja, 22 unidades que representam o resíduo mais 3 unidades que representam a resposta da primeira rede, totalizando 25, multiplicado pelo número de resíduos na janela que sempre continua igual a 17. Para a junção da saída da primeira rede com a sequência de aminoácidos, gerando então a entrada para a segunda rede, foi implementado um software, em C++Builder 5.

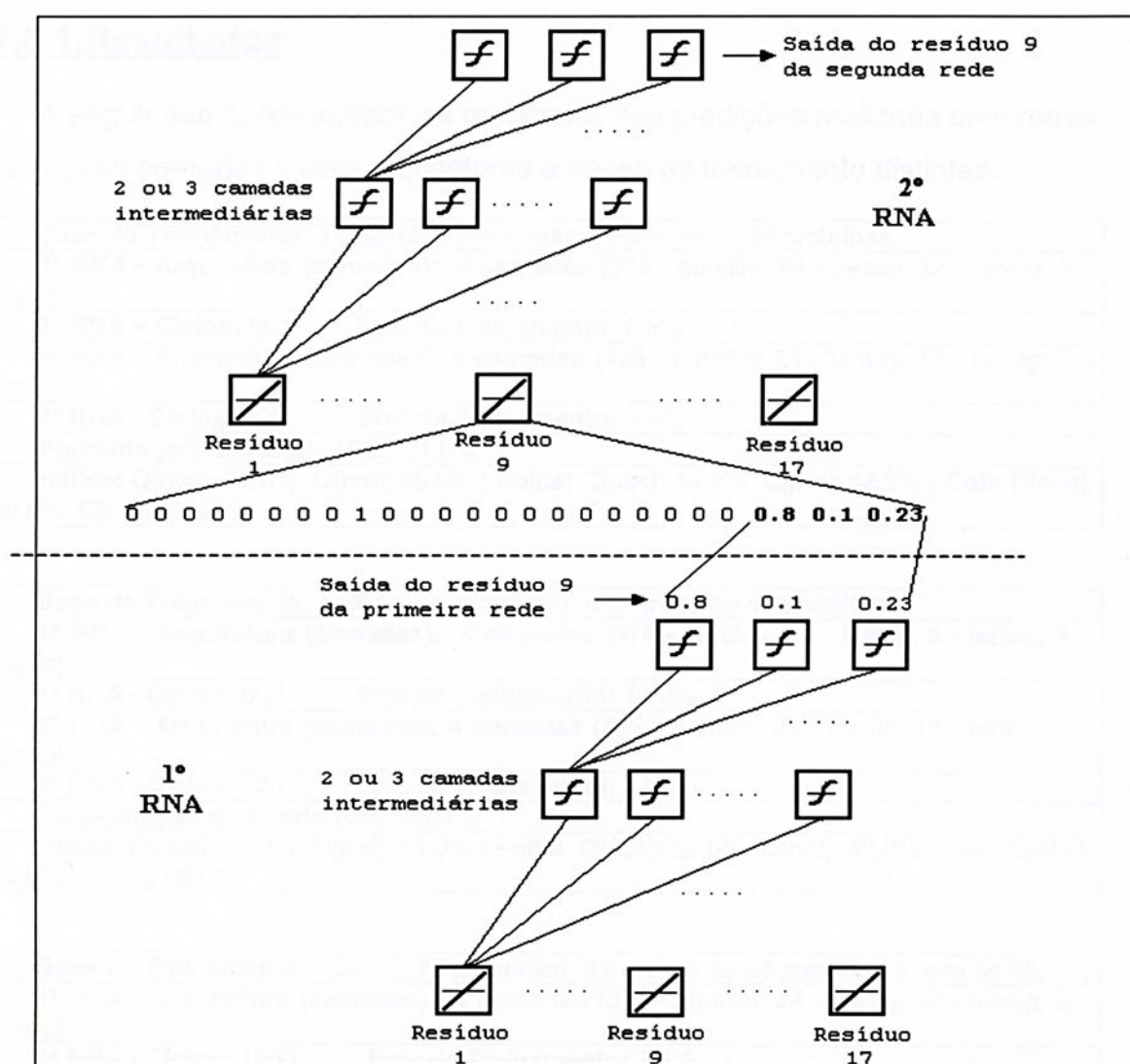


Figura 51 - Arquitetura composta de duas redes neurais. Onde a saída da primeira realimenta a segunda rede

7.6.1.Resultados

A seguir são apresentados os resultados das predições realizada com redes com duas camadas e com arquiteturas e bases de treinamento distintas.

Base de Treinamento: Todas (216 proteínas). 4 matrizes de 54 proteínas.	
1ª RNA - Arquitetura (camadas): 4 camadas (374 - purelim; 24 - tansig; 12 - tansig; 3 - logsig)	
1ª RNA - Ciclos: 800	Erro de Treinamento: 9,8%
2ª RNA - Arquitetura (camadas): 4 camadas (425 - purelim; 24 - tansig; 12 - tansig; 3 - logsig)	
2ª RNA - Ciclos: 600	Erro de Treinamento: 3,4%
Porcentagem de Acerto (Q3): 51,0%	
Hélice: Q[obs]: 38,0% Q[prd]: 55,0% ; Folha: Q[obs]: 48,0% Q[prd]: 42,0% ; Coil: Q[obs]: 59,0% Q[prd]: 65,0%	

Base de Treinamento: Todas (216 proteínas). 4 matrizes de 54 proteínas.	
1ª RNA - Arquitetura (camadas): 4 camadas (374 - purelim; 24 - tansig; 8 - tansig; 3 - logsig)	
1ª RNA - Ciclos: 600	Erro de Treinamento: 10,2%
2ª RNA - Arquitetura (camadas): 4 camadas (425 - purelim; 24 - tansig; 12 - tansig; 3 - logsig)	
2ª RNA - Ciclos: 600	Erro de Treinamento: 4,9%
Porcentagem de Acerto (Q3): 54,3%	
Hélice: Q[obs]: 42,0% Q[prd]: 54,0% ; Folha: Q[obs]: 52,0% Q[prd]: 42,0% ; Coil: Q[obs]: 61,0% Q[prd]: 66,0%	

Base de Treinamento: Hélice (173 proteínas). 2 matrizes de 54 proteínas e uma de 65.	
1ª RNA - Arquitetura (camadas): 4 camadas (374 - purelim; 24 - tansig; 12 - tansig; 3 - tansig)	
1ª RNA - Ciclos: 1000	Erro de Treinamento: 7,5%
2ª RNA - Arquitetura (camadas): 4 camadas (425 - purelim; 24 - tansig; 12 - logsig; 3 - logsig)	
2ª RNA - Ciclos: 1000	Erro de Treinamento: 2,6%
Porcentagem de Acerto (Q3): 52,1%	
Hélice: Q[obs]: 58,0% Q[prd]: 45,0% ; Folha: Q[obs]: 25,0% Q[prd]: 51,0% ; Coil: Q[obs]: 60,0% Q[prd]: 64,0%	

Base de Treinamento: Folha (56 proteínas)	
1ª RNA - Arquitetura (camadas): 4 camadas (374 - purelim; 24 - tansig; 12 - tansig; 3 - tansig)	
1ª RNA - Ciclos: 1200	Erro de Treinamento: 1,6%
2ª RNA - Arquitetura (camadas): 4 camadas (425 - purelim; 24 - tansig; 12 - logsig; 3 - logsig)	
2ª RNA - Ciclos: 400	Erro de Treinamento: 0,9%
Porcentagem de Acerto (Q3): 49,5%	
Hélice: Q[obs]: 16,0% Q[prd]: 53,0% ; Folha: Q[obs]: 67,0% Q[prd]: 37,0% ; Coil: Q[obs]: 60,0% Q[prd]: 65,0%	

Base de Treinamento: Hélice-Folha (173 proteínas)	
1ª RNA - Arquitetura (camadas): 4 camadas (374 - purelim; 24 - tansig; 8 - tansig; 3 - tansig)	

1ª RNA - Ciclos: 800	Erro de Treinamento: 6,25%
2ª RNA - Arquitetura (camadas): 4 camadas (425 - purelim; 24 – tansig; 8 - logsig; 3 – logsig)	
2ª RNA - Ciclos: 800	Erro de Treinamento: 1,5%
Porcentagem de Acerto (Q3): 55,0%	
Hélice: Q[obs]: 45,0% Q[prd]: 55,0% ; Folha: Q[obs]: 45,0% Q[prd]: 47,0% ; Coil: Q[obs]: 64,0% Q[prd]: 64,0%	

Percebe-se com estes testes, que a arquitetura com duas redes melhora em média 2% o desempenho da predição em relação à rede com apenas uma camada. A melhor porcentagem de acerto verificada com a arquitetura de uma rede foi de 52,84%. Utilizando a arquitetura formada por duas redes, chegou-se a um acerto de 55%. Dessa forma, percebe-se que mesmo mantendo a mesma base de treinamento de proteínas, pode-se melhorar a performance ao utilizar duas redes neurais em conjunto, reforçando os resultados obtidos por (Rost & Sander, 1994). As arquiteturas com duas redes neurais associadas levam uma média de 24 horas para serem treinadas em uma DELL 2.4 GHz.

7.7. Adicionando Propriedades Físico-Químicas dos Aminoácidos

No intuito de conseguir resultados cada vez mais significativos, mantendo a mesma base de dados, algumas propriedades físico-químicas dos aminoácidos foram adicionadas à entrada das redes neurais. Essas propriedades são de grande importância, uma vez que cada resíduo possui a sua, diferenciando-o dos demais aminoácidos existentes. Outro motivo que aumenta o interesse na utilização destas propriedades no treinamento das redes, é o fato que existem muitos trabalhos na literatura que fizeram uso de redes neurais em predição de estruturas 1D sem incluir essas propriedades físico-químicas (Head-Gordon, 1993) (Chandonia & Karplus, 1999).

Estudou-se a inclusão de sete diferentes propriedades dos aminoácidos na entrada das redes. A primeira foi a propensão que cada aminoácido possui em formar estruturas do tipo α -hélice, conhecida na literatura como *Helix Propensity*. Em seguida, foram adicionadas as informações de mais três propriedades, que foram: a propensidade que cada aminoácido possui em formar estruturas do tipo β -folha, nomeada por *Sheet Propensity*, e as frequências relativas de ocorrência de cada resíduo em estruturas do tipo α -

hélice e β -folha. Outras duas propriedades adicionadas foram a hidrofiliicidade e hidrofobicidade. Por fim, incluiu-se a carga do resíduo, que informa se o aminoácido é positivo, negativo ou neutro.

O principal objetivo foi verificar a diferença de desempenho entre os resultados obtidos com uma arquitetura composta de duas redes e aqueles obtidos anteriormente utilizando apenas uma RNA. As tabelas anteriores apresentam o base de treinamento utilizada, a rede utilizada (com suas funções de ativação), o erro de treinamento e o coeficiente Q3.

7.7.1. Propensão para Formação de Hélice (Helix Propensity)

Esta é uma medida de tendência de um aminoácido formar estruturas do tipo α -hélice, ou melhor, mede a quantidade de energia necessária para que o aminoácido forme uma estrutura deste tipo. Sua importância está no fato que a média global de estruturas α -hélice em uma proteína é de 30%, o que a torna o tipo mais comum de estrutura secundária encontrado. Cada resíduo tem uma propensidade diferente em formar estruturas α -hélice. Na Tabela 6 estão mostrados os valores das energias (em kcal/mol) necessárias para cada aminoácido.

Aminoáci do	P_{α}	Aminoácido	P_{α}
Arg	0,21	Ile	0,41
Ala	0,00	Phe	0,54
Cys	0,68	Tyr	0,53
Leu	0,21	Trp	0,49
Met	0,24	Thr	0,66
Glu	0,40	Gly	1,00
Gln	0,39	Ser	0,50
His	0,56	Asp	0,43
Lys	0,26	Asn	0,65
Val	0,61	Pro	0,16

Tabela 6 - Propensidade dos Aminoácidos em formar α -hélice (Creighton, 1997).

Estes valores são relativos ao aminoácido Alanina (Ala), que usualmente é o resíduo com a maior propensidade de formar estruturas α -hélice. Quanto menor for o valor do P_{α} , maior será a propensão do aminoácido em formar estruturas secundárias deste tipo e, vice-versa. Deve-se destacar ainda que os valores citados na Tabela 5 foram codificados como números binários para a sua inserção na entrada das redes, sendo os dois bits mais à esquerda para representarem a unidade e mais oito bits para representarem as duas casas decimais (quatro para a primeira casa e quatro para a segunda), sendo então dez bits para representarem os valores de *Helix Propensity* de cada aminoácido. Por exemplo, para o resíduo Valina (Val), a codificação ficaria 0001100001.

Para a inclusão desta propriedade no treinamento, foram alteradas as entradas das duas redes que compõem a arquitetura. Antes, a entrada da

primeira rede era formada por uma janela de 17 resíduos por padrão (isso não muda nunca), onde cada resíduo possuía 22 neurônios, totalizando uma entrada de 374 neurônios. Conforme mostrado na Figura 52a , após a adição do *Helix Propensity*, cada resíduo da janela passou a ter 32 neurônios, totalizando uma entrada de 544 neurônios. Já a segunda rede que possuía 425 neurônios na entrada, agora passou a ter 595 neurônios (v. Figura 52b).

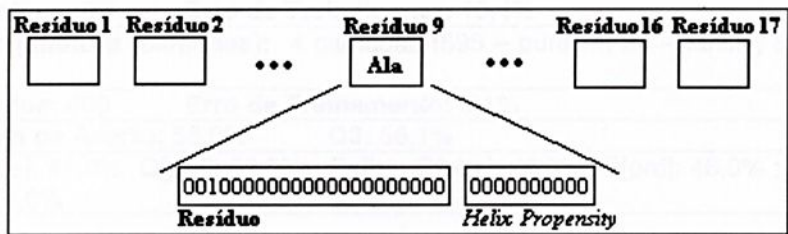


Figura 52a - Entrada da 1ª RNA com a Inclusão de *Helix Propensity*.

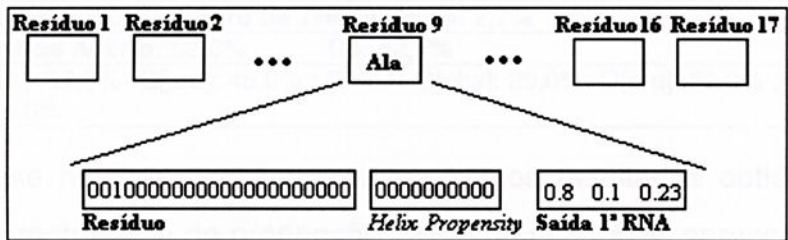


Figura 52b - Entrada da 2ª RNA com a Inclusão de *Helix Propensity*.

7.7.1.1.Resultados

A seguir são apresentados os resultados das predições obtidos com diferentes arquiteturas e com a inclusão da propriedade de propensão de formação de hélice para cada resíduo da estrutura primária. As tabelas, como nos casos anteriores, apresentam o base de treinamento utilizada, a rede utilizada (com suas funções de ativação),o erro de treinamento e o coeficiente Q3.



Base de Treinamento: Todas (216 proteínas)	
1ª RNA - Arquitetura (camadas): 4 camadas (544 - purelim; 24 - tansig; 12 - tansig; 3 - tansig)	
1ª RNA - Ciclos: 1200	Erro de Treinamento: 11,4%
2ª RNA - Arquitetura (camadas): 4 camadas (595 - purelim; 24 - tansig; 12 - tansig; 3 - logsig)	
2ª RNA - Ciclos: 1200	Erro de Treinamento: 5,7%
Porcentagem de Acerto: 55,0%	Q3: 55,1%
Hélice: Q[obs]: 38,0% Q[prd]: 60,0% ; Folha: Q[obs]: 51,0% Q[prd]: 43,0% ; Coil: Q[obs]: 66,0% Q[prd]: 66,0%	

Base de Treinamento: Todas (216 proteínas)	
1ª RNA - Arquitetura (camadas): 4 camadas (544 - purelim; 24 - tansig; 8 - tansig; 3 - tansig)	
1ª RNA - Ciclos: 400	Erro de Treinamento: 10,4%
2ª RNA - Arquitetura (camadas): 4 camadas (595 - purelim; 24 - tansig; 8 - tansig; 3 - logsig)	
2ª RNA - Ciclos: 400	Erro de Treinamento: 5,1%
Porcentagem de Acerto: 56,0%	Q3: 56,1%
Hélice: Q[obs]: 41,0% Q[prd]: 59,0% ; Folha: Q[obs]: 46,0% Q[prd]: 46,0% ; Coil: Q[obs]: 68,0% Q[prd]: 64,0%	

Base de Treinamento: Hélice (173 proteínas)	
1ª RNA - Arquitetura (camadas): 4 camadas (544 - purelim; 24 - tansig; 12 - tansig; 3 - tansig)	
1ª RNA - Ciclos: 800	Erro de Treinamento: 8,95%
2ª RNA - Arquitetura (camadas): 4 camadas (595 - purelim; 24 - tansig; 12 - tansig; 3 - logsig)	
2ª RNA - Ciclos: 1000	Erro de Treinamento: 2,7%
Porcentagem de Acerto: 52,0%	Q3: 52,1%
Hélice: Q[obs]: 58,0% Q[prd]: 45,0% ; Folha: Q[obs]: 25,0% Q[prd]: 51,0% ; Coil: Q[obs]: 60,0% Q[prd]: 64,0%	

Verificou-se no final desta simulação que os resultados obtidos, com a inclusão da característica de propensão em formar α -hélice, apresentavam um ganho de desempenho de 2% em média em relação às redes implementadas sem tal propriedade. Em particular, proteínas que possuem um grande número de α -hélices, a taxa de acerto aumentou mais do que 2%.

O próximo passo foi então incluir mais três características físico-químicas dos aminoácidos na entrada das redes neurais. As propriedades incluídas foram a propensão dos aminoácidos em formar estruturas β -folha e as frequências relativas de ocorrência dos aminoácidos em estruturas α -hélice e β -folha.

7.7.2.Propensão para Formação de Hélice e Frequências Relativas de Ocorrência

Assim como acontece na propriedade *Helix Propensity*, alguns aminoácidos têm uma tendência maior em formar estruturas β -folha do que os outros, ou seja,

a, alguns aminoácidos possuem uma propensão maior em formar estruturas β -folha do que os demais. Essa propriedade é conhecida na literatura como *Sheet Propensity* mede a quantidade de energia necessária para que o aminoácido forme uma estrutura deste tipo. A seguir estão descritos os valores (em kcal/mol) de propensão em formar β -folha para cada aminoácido (Tabela 7).

Aminoácido	P_{β}	Aminoácido	P_{β}
Arg	0,93	Ile	1,60
Ala	0,83	Phe	1,38
Cys	1,19	Tyr	1,47
Leu	1,30	Trp	1,37
Met	1,05	Thr	1,19
Glu	0,37	Gly	0,75
Gln	1,10	Ser	0,75
His	0,87	Asp	0,54
Lys	0,74	Asn	0,89
Val	1,70	Pro	0,55

Tabela 7 - Propensidade dos Aminoácidos em formar β -folha (Voet & Voet, 1995).

Estes valores são baseados em estatísticas de banco de dados e resultados experimentais de físico-química. Para incluir tais valores na entrada das redes, os mesmos foram codificados para números binários, sendo representados por nove bits, com o bit mais à esquerda usado para representar

a unidade e os demais para representarem as duas casas decimais. Usando novamente o aminoácido Valina (Val) como exemplo, a codificação ficaria 101110000.

Juntamente com a propriedade *Sheet Propensity*, foram incluídas as propriedades de freqüências relativas de ocorrência dos aminoácidos em estruturas α -hélice e β -folha. Os valores destas freqüências relativas são calculados dividindo o número de vezes que o aminoácido ocorre em tal estrutura secundária (α -hélice ou β -folha) pelo número total de elementos do aminoácido existentes no conjunto de proteínas (Stryer, 1995). Da mesma forma que acontece com o *Sheet Propensity*, estes valores são baseados em estatísticas de banco de dados e resultados experimentais de físico-química. A Tabela 8 apresenta os valores das freqüências relativas para cada aminoácido.

Aminoácido	f_{α}	f_{β}	Aminoácido	f_{α}	f_{β}
Arg	0,96	0,93	Ile	0,97	0,98
Ala	1,29	0,84	Phe	1,07	0,95
Cys	1,11	0,77	Tyr	0,72	1,15
Leu	1,30	0,80	Trp	0,99	0,94
Met	1,47	0,68	Thr	0,82	1,12
Glu	1,44	0,87	Gly	0,56	1,28
Gln	1,27	0,88	Ser	0,82	1,14
His	1,22	0,88	Asp	1,04	1,06
Lys	1,23	0,86	Asn	0,90	1,02
Val	0,91	0,98	Pro	0,52	1,27

Tabela 8 - Freqüências Relativas de Ocorrência (Strayer, 1995).

As freqüências relativas também foram codificadas para números binários. Tanto para a freqüência relativa em α -hélice quanto para β -folha são usados



nove bits para representação dos valores, sendo usados da mesma forma como acontece para representar a propriedade *Sheet Propensity*.

Em consequência à inserção destas três propriedades após o *Helix Propensity*, as entradas das duas redes que compõem a arquitetura foram novamente modificadas. Como mostrado na Figura 53a, após a adição destas novas características, cada resíduo da janela passou a ter 59 neurônios, totalizando uma entrada de 1003 neurônios. Já a segunda rede, a qual possuía 595 neurônios na entrada, passou a ter 1054 neurônios (v. Figura 53b).

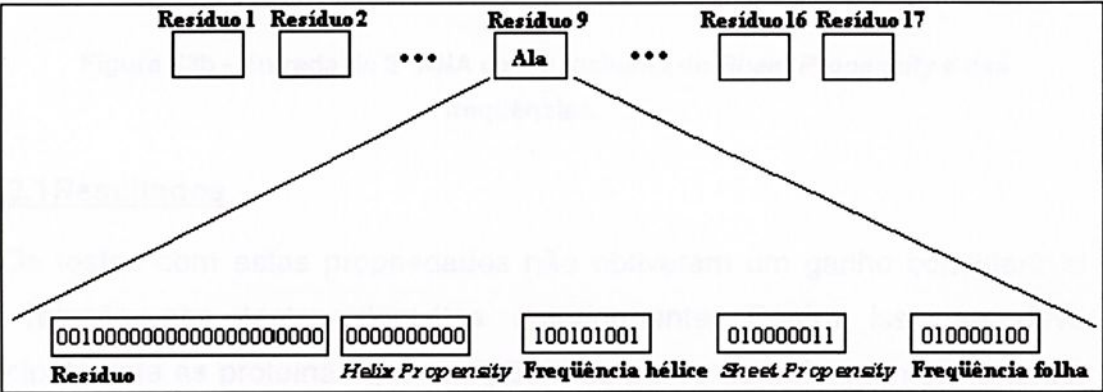


Figura 53a - Entrada da 1ª RNA com a inclusão de propensão para formação de folha e das frequências.

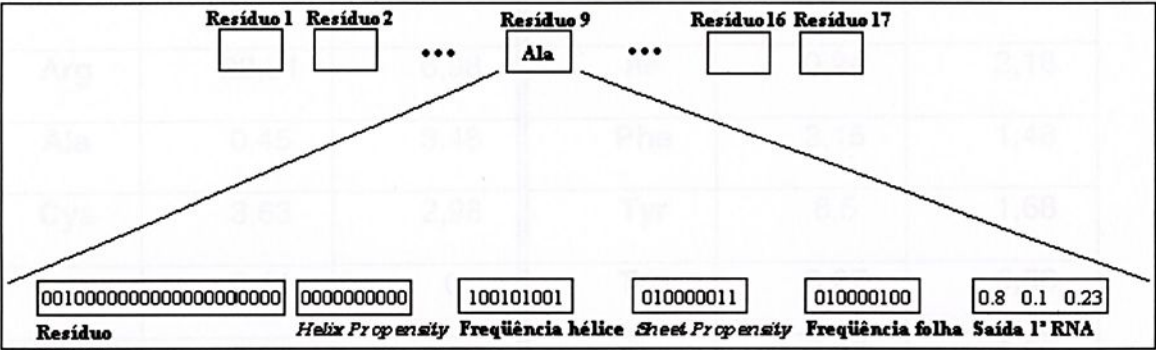


Figura 53b - Entrada da 2ª RNA com a Inclusão do *Sheet Propensity* e das frequências.

7.7.2.1 Resultados

Os testes com estas propriedades não obtiveram um ganho considerável em relação aos testes descritos anteriormente. Porém isso se deve principalmente as proteínas que compõem as bases de treinamento. Portanto, pode-se concluir que dependendo das proteínas que venham compor os banco de dados das redes, certas propriedades físico-químicas devem ou não ser utilizadas. Isso pôde ser comprovado, testando as redes para outras proteínas individuais. Além disso, pode ser que essas informações físico-químicas representem características que se sobrepõem no problema de classificação de padrões e portanto ao serem fornecidas, em conjunto, prejudicam a performance da rede.

7.7.3. Hidrofilicidade e Hidrofobicidade

Estas propriedades estão relacionadas à interação do aminoácido com a água, ou seja, se os aminoácidos gostam do meio aquoso ou não, chamadas respectivamente de hidrofilicidade e hidrofobicidade. Alguns aminoácidos têm maior tendência em “gostar” do meio aquoso por serem moléculas polares. Entretanto, a água não é um “ambiente agradável” para moléculas não-polares se comparada a outros líquidos orgânicos tais como etanol e octanol (Cre,

1997). A Tabela 9 apresenta os valores dessas propriedades (em kcal/mol) para os 20 aminoácidos, ou seja representa valores de transferência.

Aminoácido	Gostar H ₂ O	Não Gostar H ₂ O	Aminoácido	Gostar H ₂ O	Não Gostar H ₂ O
Arg	22,31	6,98	Ile	0,24	2,18
Ala	0,45	3,48	Phe	3,15	1,48
Cys	3,63	2,98	Tyr	8,5	1,68
Leu	0,11	0	Trp	8,27	0,58
Met	3,87	2,68	Thr	7,27	3,58
Glu	12,63	6,48	Gly	0	3,98
Gln	11,77	4,18	Ser	7,45	4,28
His	12,66	3,48	Asp	13,34	6,48
Lys	11,91	6,98	Asn	12,07	4,18
Val	0,40	2,48	Pro	0	2,58

Tabela 9 - Valores de Hidrofilicidade e Hidrofobicidade (Creighton, 1997).

Para sua inclusão na entrada das redes, tais valores também foram codificados para binários. Foram utilizados 14 bits para representar a Hidrofilicidade e mais 11 bits para representar as características de Hidrofobicidade dos aminoácidos (ver Figuras 54a e 54b).

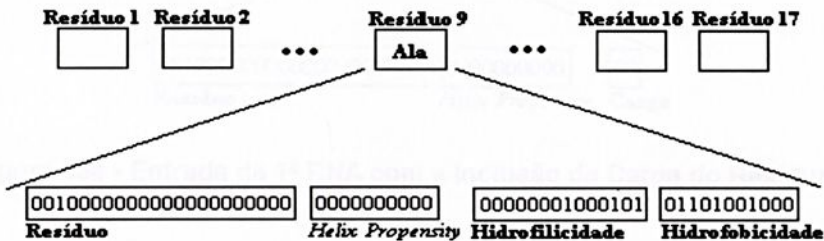


Figura 54a - Entrada da 1ª RNA com Hidrofilicidade e Hidrofobicidade.

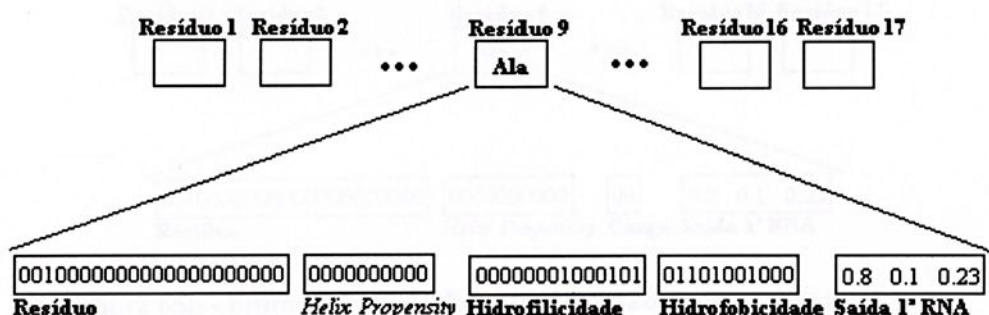


Figura 54b - Entrada da 2ª RNA com Hidrofilicidade e Hidrofobicidade

A predição utilizando as propriedades de propensão de formação de hélice, hidrofobicidade e hidrofilicidade não obteve sucesso. As porcentagens de acerto em relação às outras simulações estão bastante inferiores.

7.6.4.Carga do Resíduo

A última propriedade físico-química utilizada na entrada para treinar as redes foi a carga do resíduo. Essa característica informa se o aminoácido é positivo, negativo ou neutro, considerando o número de prótons e elétrons que este possui. Alguns aminoácidos têm prótons a mais na sua estrutura, sendo esses caracterizados como positivos. Outros têm elétrons a mais na sua estrutura, sendo considerados então como negativos. Os demais são caracterizados como neutros. Utilizou-se dois bits para representar esta propriedade. Os aminoácidos positivos foram considerados como 10, os negativos como 01 e os neutros como 00 (ver Figuras 55a e 55b).

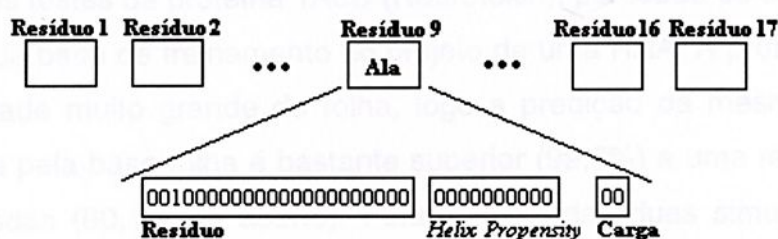


Figura 55a - Entrada da 1ª RNA com a Inclusão da Carga do Resíduo.

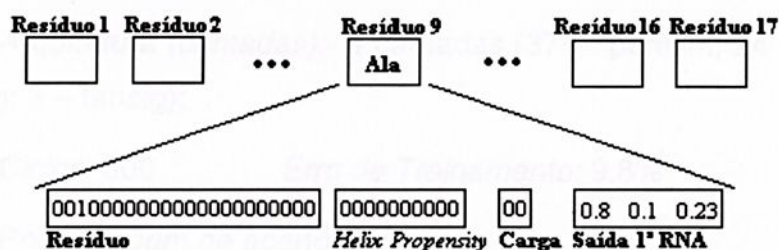


Figura 55b - Entrada da 2ª RNA com a Inclusão da Carga do Resíduo.

O objetivo do trabalho até esse momento foi investigar o limite de performance das redes na predição fazendo uso apenas de informações físico-químicas e não evolucionárias e utilizando a mesma base de dados. Pode-se perceber que o desempenho das RNAs depende em muito da combinação e codificação das propriedades bem como da arquitetura das mesmas. A seguir é apresentado um teste sobre uma proteína individual.

7.8. Teste com a proteína 1A8D (NEUROTOXIN)

Os testes realizados com esta proteína têm como objetivo avaliar o desempenho em RNAs treinadas por diferentes bases de treinamento. A proteína 1A8D, com 452 padrões (dados de entrada), possui 46% de estruturas do tipo coil, 45% de folha e 9% de hélice. Foram realizadas duas simulações com bases diferentes. Cada simulação utilizou a arquitetura de duas redes e sem informações adicionais.

Com os testes da proteína 1A8D (Neurotoxin), percebeu-se claramente a importância da base de treinamento no projeto de uma RNA. A proteína possui uma quantidade muito grande de folha, logo a predição da mesma em uma rede treinada pela base folha é bastante superior (99,3%) a uma rede treinada pela base todas (60,1% de acerto). Foram realizadas duas simulações com bases diferentes. Cada simulação utilizou a arquitetura de duas redes e sem informações adicionais.

7.8.1 Simulação 1

1ª RNA

- Base de treinamento: Todas.

- *Arquitetura (camadas):* 4 camadas (374 - purelim; 24 - tansig; 12 - tansig; 3 – tansig);
- *Ciclos:* 800 *Erro de Treinamento:* 9,8%
- *Porcentagem de acerto:* 53,4%

	HÉLICE	FOLHA	COIL
No.	38	202	213
Observado			
Nro preditos	89	105	200
Q (obs)	44 %	43 %	64%
Q (prd)	19 %	82 %	69 %

2ª RNA:

- *Arquitetura (camadas):* 4 camadas (425 - purelim; 24 - tansig; 12 - tansig; 3 – tansig);
- *Ciclos:* 600 *Erro de Treinamento:* 3,47%
- *Porcentagem de acerto:* 60,1%

	HÉLICE	FOLHA	COIL
No observado	38	202	213
Nro preditos	91	172	189
Q (obs)	57 %	61 %	59%
Q (prd)	24 %	72 %	67%

7.8.2 Simulação 2

1ª RNA

- *Base de Treinamento:* Folha (56 proteínas);
- *Arquitetura (camadas):* 4 camadas (374 - purelim; 24 - tansig; 12 - tansig; 3 – tansig);
- *Ciclos:* 1200 *Erro de Treinamento:* 1,6%
- *Porcentagem de acerto:* 97,3%

	HÉLICE	FOLHA	COIL
<i>No. Observado</i>	38	202	213
Nro preditos	31	209	213
Q (obs)	76 %	99%	99%
Q (prd)	93 %	95 %	99%

2ª RNA

- *Arquitetura (camadas):* 4 camadas (425 - purelim; 24 - tansig; 12 - tansig; 3 – tansig);
- *Ciclos:* 400 *Erro de Treinamento:* 0,9%
- *Porcentagem de acerto:* 99,3%



	HÉLICE	FOLHA	COIL
No observado	38	202	213
Nro preditos	38	202	213
Q (obs)	97 %	99 %	99%
Q (prd)	97 %	99 %	99%

Os testes realizados com esta proteína comprovaram a importância tanto de redes treinadas pela base folha. O software Conversor possui uma propriedade importante que é fornecer qual RNA se enquadra na predição de determinada proteína, considerando suas características de estrutura.

Os resultados apresentados e discutidos nesse capítulo reforçam a idéia defendida por (Head-Gordon, 1993) de que o projeto da arquitetura de uma rede neural é tão importante quanto o da base de dados para a performance dela na predição 1D. Uma rede projetada com muito cuidado e com entradas adequadamente codificadas pode suprir as dificuldades de um base de dados pobre ou pequena e melhorar, de forma significativa, o desempenho da rede neural. Também é preciso tomar muito cuidado com o tipo de dados fornecido como entrada para a rede, para que não haja padrões se sobrepondo, o que dificulta o problema de classificação de padrões. O treinamento de algumas redes neurais chegou a levar aproximadamente 24 horas em um Pentium IV. Já os testes e operacionalização das redes, após serem treinadas, é praticamente instantâneo.

Discussões e Conclusões

8.1.Considerações Finais

As proteínas são moléculas complexas que possuem uma estrutura terciária (tridimensional) específica e que realizam tarefas de extrema importância para o organismo, como a catálise de reações químicas, transporte, reconhecimento e transmissão de sinal. A função das proteínas conhecidas está determinada pela sua estrutura espacial. Dessa forma, obter a estrutura tridimensional da proteína no seu estado nativo, e compreender as forças que a estabilizam são um dos problemas fundamentais da Biofísica de proteínas.

A predição de estrutura tem por objetivo investigar e propor ferramentas que permitam prever a estrutura secundária e terciária de proteínas a partir de sua seqüência primária e de outras informações. Até o momento, em geral, não possuímos habilidade suficiente para prever a estrutura terciária (tridimensional) de uma proteína na grande maioria, a partir de sua seqüência (estrutura primária). Porém a utilização de técnicas como redes neurais artificiais tem permitido bons e interessantes resultados na predição da estrutura secundária.

O uso de algoritmos genéticos no estudo das estruturas terciárias de peptídeos e pequenas proteínas também tem apresentado resultados interessantes e promissores. Há fatores extremamente relacionados com o algoritmo genético e sua eficiência que devem ser explorados como: *os critérios para selecionar os cromossomos que irão participar da recombinação e critérios para replicar cromossomos, tamanho da população e taxa de mutação*. Pode-se verificar que a inclusão de um termo, de solvatação proporcional à área acessível, afeta e melhora sensivelmente o desempenho dos algoritmos genéticos que eles são uma ferramenta. O uso de bibliotecas de

rotâmeros também se mostrou um recurso interessante de ser utilizado em problemas de predição de estruturas e design.

Numerosos trabalhos utilizando redes neurais artificiais para estudar estruturas protéicas (primária, secundária e terciária) têm sido realizados e descritos na literatura. A predição de estruturas secundárias é um passo útil e importante para compreender como a seqüência de aminoácidos de umas proteínas determina o seu estado nativo.

Os algoritmos de predição de estruturas secundárias com melhores resultados, no momento, são baseados em redes neurais. A maioria dos métodos, aplicando RNAs, correntemente disponíveis possuem três estados de predição e alcançam uma performance de 72% a 78% de forma geral, podendo ser melhor dependendo da base de dados e das informações de entrada da rede neural.

Entre as possíveis aplicações de RNAs no estudo de estruturas de moléculas como proteínas estão:

- Predição de estruturas secundárias através classificação/reconhecimento de padrões;
- Predição de estruturas terciárias de proteínas através de otimização de uma função potencial de energia;
- Predição de possíveis seqüências de aminoácidos para uma dada proteína de forma a obter as conformações de mais baixa energia.
- Predição de seqüências de aminoácidos (estruturas primárias) que levem a uma estrutura secundaria ou terciária desejada.

Outros aspectos de estruturas de proteínas, tal como a classe estrutural, também pode ser predita utilizando redes neurais. Podem-se utilizar redes neurais para associar as proteínas a uma das grande quatro classes (toda α , toda β , α/β e outras) com uma precisão que pode chegar a 78% em alguns trabalhos.

Um dos trabalhos pioneiros de aplicação de RNA na predição de estrutura é devido a Holley and Karplus (Holley e Karplus, 1991). Eles usaram uma rede



MLP, não recorrente para predizer elementos de estrutura secundária de proteínas a partir da seqüência de resíduo. Eles codificam os dados de entrada em janelas de resíduos adjacentes e para cada resíduo, usaram 21 entradas binárias (ou seja, assumiram apenas zero e um), sendo que apenas uma das 20 entradas estava ativada (cada entrada representa um tipo de aminoácido) e uma codificada quando a janela sobrepunha o fim da cadeia.

A rede utilizada possuía uma camada intermediária com duas unidades e uma camada de saída também com duas unidades, que representava/codificava uma estrutura secundária associada ao resíduo central na janela. Para facilitar a comparação com outros métodos de predição, Holley & Karplus adotaram três tipos de estrutura secundária: hélice, folha e *coil* (Holley & Karplus, 1991).

O conjunto de dados utilizados para treinar e testar a rede era composto de 62 proteínas, sendo que o conjunto de treinamento formado por 48 proteínas e o conjunto de teste (para predição) por 14 proteínas. Holley & Karplus testaram vários tamanhos de janela e, nesse caso, a janela de tamanho 17 mostrou os melhores resultados. Também foram testadas redes neurais com diferentes tamanhos de cadeia intermediária (variando de 2 a 20). Apesar da rede com 20 unidades (neurônios) na camada intermediária ter apresentado o melhor resultado para o conjunto de treinamento; a rede com duas unidades na camada intermediária mostrou o melhor resultado para o conjunto de teste. A precisão obtida para janelas de tamanho 17 e redes com dois neurônios em sua camada escondida foi de 63.2% para o conjunto de teste e 68.5% para o conjunto de treinamento.

Qian and Sejnowski também utilizaram redes neurais não lineares para predizer estrutura secundária de proteína globulares e avaliaram o efeito de ruído nos dados de treinamento em relação à curva de aprendizado da rede e sua performance nos teste (Qian & Sejnowski, 1996).

Keneller et al. aplicaram uma rede neural *MLP* com duas camadas e utilizaram uma janela de 13 resíduos adjacentes para investigar a predição de estruturas secundárias. Eles foram os primeiros a utilizar uma janela de resíduos adjacentes e predizer a estrutura do resíduo central (Keneller, 1990).



A camada de saída da rede possuía três neurônios indicando: hélice, folha ou *coil*. Como entrada era fornecida apenas a seqüência de aminoácidos da janela.

Chandonia and Karplus aplicaram duas redes neurais denominadas primária e secundária para prever estruturas secundárias e classes estruturais ou classes de estrutura (Chandonia & Karplus, 1999). Nesse estudo foi utilizado um conjunto de 681 proteínas com estruturas disponíveis no *Protein Data Bank* (PDB). A rede neural primária utilizada para prever a estrutura secundária era similar a várias descritas por Karplus em trabalhos anteriores (Chandonia & Karplus, 1996). Nesse caso, a rede utilizava uma janela para prever, iterativamente, a estrutura secundária para cada resíduo da proteína. Para cada resíduo na janela de entrada, o tipo do resíduo era codificado e apresentado para a rede em 21 unidades de nível de entrada que representavam o tipo de aminoácido que aparecia naquela posição da estrutura primária e uma possível sobreposição da janela com o fim da cadeia. A saída da rede consistia de duas unidades que correspondiam à predição de α -hélice e β -folha respectivamente e assumiam valor zero ou um. A rede neural (secundária) foi utilizada para refinar os resultados produzidos pela primeira rede; classificando a proteína em classes : toda α -hélice, toda β -folha, α/β e outras. Nessa rede, os dados de entrada eram a codificação hélice ou folha para cada resíduo, obtida como resultado da predição realizada pela primeira rede neural.

Kono et. al (Kono et al., 1994) descreveram o uso de uma rede de autômatos para a predição de seqüência e a conformação de cadeias laterais a partir da geometria da cadeia principal. Proteínas globulares possuem um empacotamento altamente denso como resultado das cadeias laterais dos resíduos do núcleo hidrofóbico. Nesse contexto, Kono et. al apresentaram um método para minimização de energia utilizando uma rede de autômatos, muito semelhante a uma rede de Hopfield, para prever tanto a seqüência de aminoácidos quanto a geometria das cadeias laterais dos resíduos que formam o core da proteína. Eles utilizaram uma biblioteca de rotâmeros, sendo que as coordenadas do esqueleto (backbone) eram fornecidas e mantidas fixas de acordo com os dados da proteína armazenados no PDB (Kono et al., 1994).



Nesse trabalho, foram usados seis tipos de resíduos hidrofóbicos apenas para prever o core das proteínas (ILE, LEU, MET, ALA, PHE e VAL) já que as regiões de núcleo são compostas por resíduos hidrofóbicos. O método foi aplicado para prever o core de três proteínas: *citocromo b₅₆₂*, *barnase* e *isomerase*.

A Biblioteca de rotâmeros utilizada por Kono et al. foi definida por Ponder e Richards para reduzir o grau de liberdade dos rotâmeros (Ponder & Richards, 1987). Nesse método, um autômato é associado para cada posição do rotâmero, semelhante a associar um rotâmero para cada neurônio em uma rede de Hopfield e minimizar a função de energia da rede neural. As cadeias laterais possuem um papel fundamental na estrutura da proteína, o que torna importante investigar o estudo de técnicas como redes neurais utilizando bibliotecas de rotâmeros na predição de estruturas de proteínas.

Rost & Sander (Rost & Sander, 1993) aplicaram rede neural MLP, com duas camadas, sobre uma base de dados não redundante de 130 cadeias de proteínas para prever a estrutura secundária de proteínas. Um elemento chave nesse trabalho de 1993 foi o uso de informações evolucionárias obtida por alinhamentos múltiplos de sequência que foram utilizadas como entrada ao invés de seqüências simples. Esse foi um dos primeiros trabalhos a incluir informação evolucionária para auxiliar as redes neurais na predição de estruturas e apresentou um aumento de 6% a 8% na capacidade da predição. A combinação de três níveis de redes resultou em uma taxa de acerto de 70.8% para proteínas globulares.

O sistema de rede neural utilizado por Rost & Sander consistia de três níveis, forma semelhante ao PHD. O primeiro nível classificava "*strings*" de resíduos adjacentes em três classes de estrutura secundária: hélice, folha e *turn*. Para cada resíduo a frequência de ocorrência de cada um dos 20 aminoácidos em uma posição específica no alinhamento múltiplo era computada.

Esses vinte números correspondem a uma célula básica da camada de entrada de dados da rede neural, isto é, para cada resíduo de uma janela de tamanho $w=13$ havia vinte unidades de entrada que correspondiam à



porcentagem de ocorrência de cada tipo de aminoácido naquela posição correspondente.

A saída correspondia à classe de estrutura secundária do resíduo central e as entradas e saídas foram codificadas em números binários zeros ou um. O primeiro nível da rede foi treinado para classificar segmentos de resíduos mutuamente independentes em termos do estado de um único resíduo central. A segunda rede levava em conta a correlação entre os segmentos. Nessa rede, a entrada era uma janela de 17 resíduos adjacentes e em um terceiro nível foi aplicado um júri na saída de diferentes redes, computando-se a média aritmética para 8 redes neurais diferentes. Rost e Sanders treinaram as duas redes de forma balanceada e não balanceada, gerando 2X2 redes treinadas independentes e distintas em relação ao conjunto de treinamento (Rost & Sander, 1994).

No artigo de 1994, eles (Rost & Sander, 1994) estenderam o trabalho anterior adicionando informações derivadas do alinhamento múltiplo como dado de entrada para a rede neural. Eles utilizaram o peso de conservação relativo a posições específicas da seqüência para aumentar a performance e o número de inserções e remoções para evitar sobrepredição e melhorar a exatidão. A rede final obteve uma exatidão de 71.6% em um teste *cross-validation* sobre um conjunto de 126 proteínas. Nesse trabalho, Rost e Sander utilizaram também dois níveis de rede e uma camada de júri. As redes neurais foram treinadas sete vezes sobre conjuntos diferentes de 108 cadeias de proteínas e testada sobre as 18 seqüências que ficaram fora do treinamento (ROST & SANDER 1994). As redes utilizadas foram do tipo *multilayer perceptron* e treinadas com algoritmo de *retropropagação*. A primeira rede realizava uma classificação “seqüência-estrutura”. Os dados de entrada das duas redes neurais e a forma de codificá-los bem como saída das redes neurais são idênticos a do PHD.

A Segunda rede, no segundo nível, realizava uma classificação “estrutura-estrutura”, como caso do PHD, refinando o resultado obtido pela primeira rede neural. Essa rede possuía como entrada as estruturas secundária previstas pela primeira rede para cada resíduo e fornecia como saída, novamente, a estrutura secundária do resíduo central da janela. A única diferença estava no



tamanho da janela de resíduos adjacentes utilizada pela rede neural. Nesse trabalho, Rost & Sanders utilizaram uma janela de 17 resíduos adjacentes para a Segunda rede neural ao invés de 13.

Cai et. al. descreveram uma proposta de aplicação de redes de Kohonem para predição de dobras- β que são componentes importantes para a estrutura de uma proteína. Esse tipo de estrutura, geralmente, é composto de quatro resíduos consecutivos R_i , R_{i+1} , R_{i+2} e R_{i+3} e possibilita que a cadeia polipeptídica dobre-se 180° sobre si mesma. As dobras- β são estabilizadas por pontes de hidrogênio entre a cadeia principal, ou seja, entre o grupo $C=O$ do resíduo R_i e o grupo NH do resíduo R_{i+3} e a distância entre o C_i^α e o C_{i+3}^α é menor que sete (Cai et al., 2000). As voltas- β (β -Turns) formam um dos tipos mais comuns de estruturas não repetitivas nas proteínas. O número de resíduos em voltas- β correspondem em média a 25% do total de resíduos. Nos últimos tempos, tem-se mostrado a importância estrutural e funcional desse tipo de estrutura (Cai et al., 2000).

Cai et al. utilizaram redes neurais para prever o domínio estrutural de uma proteína em sete classes: toda α , toda β , α/β , $\alpha+\beta$, multi-domínios, proteínas pequenas e peptídeos (Cai et al., 2000). A classe estrutural de um domínio de proteína está relacionada com a sua composição de aminoácidos. Muito esforço tem sido feito no sentido de tentar prever o domínio de uma proteína a partir de sua composição de aminoácidos (seqüência primária). Eles utilizaram uma rede neural do tipo Kohonen auto-organizáveis treinadas e testadas sobre a base de dados SCOP e obtiveram uma exatidão nas predições de domínio em torno de 83.7% - 89.9%.

Head-Gordon discutiu que o sucesso da aplicação de redes neurais na predição de estrutura secundária está limitada por três problemas principais: (i) as base de dados esparsas; (ii) um projeto pobre das arquiteturas, assim como uma codificação ruim dos dados de entrada para as redes neurais; e (iii) o fato de termos que resolver um problema de otimização global em um espaço de múltiplos mínimos das variáveis das redes neurais (Head-Gordon, 1993). Nesse contexto, Head-Gordon discutiram como melhorar a performance das redes neurais através de um projeto mais adequado de suas arquiteturas.

Head-Gordon aplicaram as redes neurais em modelos de peptídeos constituídos de tipos de aminoácidos (modelo hp).

Head-Gordon e seu grupo trabalharam com redes neurais explorando como um bom projeto de arquitetura da rede pode aumentar o rendimento delas. Em alguns trabalhos eles utilizaram como dados de entrada a propensão para iniciar, continuar e terminar uma hélice, bem como, a propensão para formarem folha- β e a hidrofobicidade relativa de cada resíduo. Crivelli et al apresentavam uma estratégia de otimização global para a predição da estrutura terciária de proteínas, na qual as redes neurais foram utilizadas para predizer a estrutura secundária que foi utilizada como restrições na otimização (Crivelli et al., 2000).

Recentemente Cuff e Barton avaliaram de forma mais detalhada como o uso de tipos diferentes de perfis (*profiles*) de alinhamentos múltiplos, a partir das mesmas seqüências, pode melhorar a performance das redes neurais (Cuff & Barton, 1999).

Nesse trabalho, Cuff e Barton exploraram e avaliaram como o uso, mais sofisticado, do alinhamento múltiplo e das informações podem ser importantes para melhorar o rendimento das redes neurais na predição de estrutura secundárias e propuseram um novo método baseado em redes neurais. Nesse novo método, as informações de alinhamento são mais exploradas do que na maioria dos métodos de 3ª geração (Cuff & Barton, 1999). Para cada uma das 480 proteínas do conjunto de treinamento, um alinhamento múltiplo foi realizado utilizando o Blast e o PSIBLAST e os alinhamentos foram representados como perfis (*profiles*) para as Redes Neurais. A janela de resíduos adjacentes utilizada pela rede possuía tamanho 17. Os autores obtiveram uma taxa de acerto de até 76.4% para a predição de estruturas secundárias nesse trabalho (Cuff & Barton, 1999).

Cuff and Barton projetaram uma base de dados de 369 proteínas para avaliar o rendimento dos algoritmos de predição propostos na literatura: DSC, PHD, NNSSP e PREDATOR, avaliando a taxa de acerto desses métodos (Cuff & Barton, 2000). Rabow and Scheraga descreveram uma aplicação de redes neurais para predição de estruturas dentro de uma rede cúbica. Define-se uma



função potencial e a rede neural faz uma busca por conformações de mais baixa energia dentro da rede cúbica. Rabow e Scheraga compararam os resultados obtidos com os resultados obtidos pelo método de Monte Carlo (Rabow & Scheraga, 1993). Nesse trabalho, Scheraga e Rabow obtiveram resultados melhores com as redes neurais comparado com o método de Monte Carlo. Muskal e Kim também investigaram o uso de redes neurais do tipo MLP na predição de estruturas secundárias (Muskal & Kim, 1992).

As Redes Neurais Artificiais são ótimas ferramentas para classificação e reconhecimento de padrões. Portanto são boas ferramentas para predição 1D. Pesquisadores, inicialmente, aplicavam redes neurais como caixas pretas e buscavam uma melhora na performance por meio da otimização de parâmetros, como: passo de aprendizado, arquitetura da rede e a função de ativação. Hoje, há uma necessidade de compreender melhor a rede neural ao aplicá-la na predição de estrutura de proteínas. Isso que pode possibilitar um aumento na exatidão das predições.

Diferentes estratégias têm sido implementadas para melhorar a performance das redes na predição de estruturas secundárias de proteínas. Essas estratégias podem ser agrupadas em cinco grupos principais:

- Adicionar novos tipos de informações biológicas (como: informações evolucionárias, informações sobre a cadeia principal da molécula e dados sobre contatos).
- Alteração da forma como a informação é apresentada para a Rede Neural. (i.e., estratégias que modificam o conjunto de treinamentos).
- Utilizar pós-processamento / filtros antes da predição
- Alterar a arquitetura *feed-forward* padrão utilizada.
- Treinamento Balanceado

O treinamento balanceado é um conceito dos métodos da 3ª geração. A distribuição de exemplos de treinamento é desigual, observa-se que 32% dos resíduos estão em hélice, 21% são folhas e observa-se que 47% são *loop ou coil*. Dessa forma se escolhermos, a cada passo de treinamento, um resíduo de

cada classe para ser predito (i.e., uma janela com resíduo central em hélice, uma janela com resíduo central em folha e outra com o resíduo central e coil), o treinamento balanceado poderá aumentar a exatidão do treinamento.

Todas essas estratégias devem melhorar, significativamente, a performance das Redes Neurais na predição de estruturas de proteínas e podem ser combinadas e utilizadas em conjunto. Há diferentes linhas quando se discute como melhorar a performance das redes neurais na predição de estrutura de proteínas. Alguns autores exploram o problema de melhorar a base de dados, outros autores exploram também o uso de informações evolutivas obtidas pelo alinhamento de seqüências e outros defendem, ainda, que um bom projeto da rede é um fator chave para a performance da predição. Na realidade todo esforço em qualquer uma dessas três direções é necessário para melhorar a performance das redes neurais na predição de estruturas.

Nessa Tese, foram explorados os efeitos do projeto da arquitetura e o uso de várias propriedades físico-químicas (isoladas e combinadas) no desempenho das redes neurais na predição 1D. Comprovando que tanto o projeto da base de dados, bem como o projeto da arquitetura da rede neural são extremamente importantes para o desempenho da predição 1D usando RNAs.

Como dito anteriormente, algoritmos genéticos oferecem uma forma eficaz de busca conformacional. Nesse sentido, na última década, houve um significativo esforço para aprimorar os algoritmos para busca conformacional de peptídeos, hidrocarbonetos e proteínas.

Friedrichs e Wolynes utilizaram algoritmo genético para prever a estrutura de uma proteína em um modelo de rede cúbica simplificado, no qual cada aminoácido era representado pelo seu Carbono alfa - C_{α} (Friedrichs & Wolynes, 1996).

Tuffery aplicou AGs para o problema de empacotamento ou determinação da conformação das cadeias laterais de cada aminoácido com as coordenadas dos átomos da cadeia principal fixos. Nessa abordagem, os cromossomos representam uma conformação específica da proteína, ou seja, um conjunto de rotâmeros (Tuffery et al., 1991). Le Grand e Merz aplicaram algoritmos



genéticos para busca conformacional de vários peptídeos e da crambina com 46 aminoácidos ao total (Le Grand & Merz, 1993) (Le Grand & Merz, 1994).

Judson et al. compararam a capacidade de algoritmo genéticos, recozimento simulado (*simulated annealing*) e busca randômica na tarefa de localizar conformações com mínimos globais para polímeros com tamanhos variados, em redes bidimensionais (Judson et al. 1992). Unger and Moulton compararam o desempenho de algoritmos genéticos com o método de Monte Carlo na simulação de folding em uma rede bidimensional (Unger & Moulton, 1993). Esse trabalho mostrou que o algoritmo genético foi superior ao método de Monte Carlo, em relação aos resultados obtidos para o problema de busca conformacional.

Pedersen e Moulton aplicaram algoritmos genéticos para determinar a estrutura de uma proteína a partir da sequência primária. Eles utilizaram AGs para estudar a estrutura de 28 fragmentos de proteínas. Nesse trabalho a função de energia potencial é composta de fatores: (i) contribuição da energia eletrostática de cada resíduo com resíduos próximos na sequência; (ii) contribuição de grupos polares (pontes de hidrogênio) e (iii) um termo de solvatação (Pedersen & Moulton, 1997). Cada cromossomo representa uma conformação e é composto de pares de ângulos ϕ , ψ e χ obtidos a partir de uma biblioteca de 255 estruturas de proteínas não redundantes. Nesse trabalho, Pedersen e Moulton utilizam apenas o operador de cruzamento e não aplicam o operador mutação. Em cada nova geração, dez por cento dos cromossomos de energia mais baixa são conservados, ou seja, são automaticamente passados (replicados) para a nova geração. Os pares de cromossomos e a posição em que será realizado o cruzamento possuem uma probabilidade de serem escolhidos dependentes da energia. De forma, que os cromossomos com menor energia passem para a próxima geração.

O trabalho descreve os resultados obtidos, na predição de parte da estrutura da proteína Barnase, de vários peptídeos e de fragmentos de proteínas. Pedersen e Moulton obtiveram para 18, de um total de 28 moléculas examinadas, conformações de energias baixas e com o RMS (sobre os carbonos alfa) menor que 3 Angstroms comparado com estruturas obtidas através de métodos experimentais. Para os dez casos (moléculas ou

fragmentos de moléculas) em que não foi encontrada uma estrutura de baixa energia e com um RMS baixo, existem três possíveis respostas: (i) a busca conformacional não converge para um mínimo global de energia; (ii) o campo de força utilizado possui vários mínimos falsos para essas moléculas (fragmentos) e (iii) as estruturas preferidas desses fragmentos não são independentes do contexto. Pendersen e Moult também procuraram avaliar o papel do efeito hidrofóbico na predição das estruturas. O campo de força utilizado pelos autores era composto de três termos: um para descrever as interações eletrostáticas da cadeia principal; um para considerar outras interações eletrostáticas intramoleculares e outro termo para descrever a interação de solvatação.

Dandekar e Argos utilizaram algoritmos genéticos para modelar o enovelamento dos carbonos alfa (C_α) da cadeia principal (*backbone*) de uma proteína (Dandekar & Argos, 1991, 1998). Nesses trabalhos, cada cromossomo codifica uma conformação de uma proteína através de pares de ângulos (ϕ , φ), permitindo que cada par assuma um valor num conjunto de possíveis valores discretos determinados entre -180 e $+180$. Dandekar e Argos representaram a cadeia lateral por uma esfera de $1.9 \text{ \AA}$. A função de energia potencial utilizada possui contribuições de pontes de hidrogênio, efeito hidrofóbico e preferência de estruturas secundárias e conseguiram realizar, com sucesso, a predição de trechos de proteínas.

Sun et al. descreveram a cadeia principal da proteína foi descrita completamente e as cadeias laterais foram representadas por um átomo virtual e utilizaram um campo de força mais completo do que o utilizado por Pendersen e Moult. O método obteve boas estruturas para peptídeos e proteínas pequenas. Os valores de RMS, em relação as estruturas depositadas no PDB (sobre o carbono alfa), ficaram menor do que $3.0 \text{ \AA}$ (Sun, 1995). Eles também obtiveram bons resultados na predição de trechos de proteínas e peptídeos.

Le Grand & Merz, em uma revisão, discutiram o uso de algoritmos genéticos para predição de estruturas terciárias. Nesse trabalho, cada cromossomo representa uma conformação através de pares de ângulos Phi e Psi. A população inicial é gerada randomicamente e os pares de cromossomos são



selecionados para o cruzamento a partir de uma probabilidade que depende da sua energia (Le Grand & Merz, 1994).

Konig e Dandekar discutiram o uso de algoritmos genéticos para estudar o problema de enovelamento em uma rede bidimensional e apresentavam uma variante para o operador cruzamento sistemático que melhora a performance dos algoritmos. Eles compararam o algoritmo genético padrão com o implementado com cruzamento sistemático para diferentes tamanhos de cadeias e mostraram que esse último encontra mínimos de energia com valores melhores, além de ser mais rápido (Konig & Dandekar, 1999).

No cruzamento sistemático, a cada geração, um conjunto dos melhores indivíduos é selecionado para realizar o cruzamento. Então, para cada par de cromossomos desse conjunto, todas as posições possíveis para o cruzamento são testadas. A posição que originar os filhos com menor energia é selecionada.

Recentemente, Cui et al. descreveram um algoritmo para calcular a estrutura nativa a partir de sua sequência primária (Cui et al., 1998). Os autores utilizaram um potencial de solvatação proporcional à área, não polar, acessível ao solvente. Eles também realizaram a predição de superestruturas secundárias e utilizaram os resultados como restrições para o processo de busca conformacional. O algoritmo genético foi aplicado às proteínas e peptídeos e obtiveram como resultados estruturas com o RMS (sobre o carbono alfa) menor 2.0 Angstroms para alguns peptídeos.

Dejarlais & Handel desenvolveram um sistema computacional para design e predição de núcleos hidrofóbicos de proteínas que inclui a descrição da cadeia principal explícita. O software combina algoritmo genético e o método de monte carlo em duas etapas. A estrutura do backbone é avaliada pela função de energia ou por restrições potenciais que enfatizam a preservação da geometria local (Dejarlais & Handel, 1999).

Segundo eles, assumir o backbone fixo e trabalhar com as cadeias laterais representadas nos cromossomos é extremamente poderoso, pois reduz drasticamente a complexidade do problema e o tempo de processamento.



Porém, a rigidez imposta pelo esqueleto fixo pode impedir o projeto de estruturas para o núcleo que se mostram, experimentalmente, viáveis.

Dsejarlais & Handel descreveram o desenvolvimento e caracterização de uma primeira geração de programas para otimização e simulação simultânea da estrutura e seqüência de núcleos hidrofóbicos. O método foi aplicado em uma série de núcleos hidrofóbicos variantes de phage T4 lysozyme (Desjarlais & Handel, 1999). Os resultados evidenciam que o uso do modelo da cadeia principal fixa leva a uma boa correlação entre a estabilidade calculada e determinada experimentalmente, mostrando que as cadeias laterais conferem uma importante e significativa flexibilidade. Os autores utilizaram uma população de 200 cromossomos gerada, inicialmente, de forma randômica. As cadeias laterais e o backbone são codificados (representados) nos cromossomos.

Os autores desenvolveram duas versões diferentes do programa, uma na qual o backbone era fixa; e outra, onde a flexibilidade foi introduzida no backbone somando valores entre -7° e $+7^\circ$ nos ângulos ϕ e ψ originais da molécula. A incorporação da flexibilidade na cadeia principal do núcleo hidrofóbico influenciou tanto na estabilidade calculada para novos núcleos projetados, como na predição da estrutura desses. A investigação do papel do empacotamento das cadeias laterais no enovelamento protéico e na estabilidade da estrutura terciária de uma proteína é importante para os métodos de predição, como algoritmos genéticos, e tem sido investigado (Hurley, 1994).

O uso de algoritmo genético tanto para a predição como para o design de proteínas pequenas, peptídeos e pedaços de proteínas tem sido investigado por diversos grupos de pesquisas. Esses algoritmos têm-se revelado uma técnica eficiente e interessante tanto para o problema de design quanto para o de predição. Há vários fatores a serem investigados nesse tipo de problema, como por exemplo: o quanto é vantajoso ou não utilizar bibliotecas de rotâmeros; se há grandes diferenças entre os resultados obtidos com bibliotecas de rotâmeros dependentes do backbone e não dependentes da cadeia principal e a influência do potencial de solvatação na eficiência do algoritmo genético.



No trabalho de doutorado, descrito nessa Tese, foi investigado o uso de bibliotecas de rotâmeros dependentes e não dependentes em conjunto com algoritmo genético na predição da estrutura terciária de peptídeos e núcleos hidrofóbicos de proteínas globulares bem como na projeção desses núcleos. O objetivo principal foi comprovar a viabilidade e eficácia do uso de bibliotecas de rotâmeros em conjunto com algoritmos genéticos na predição 3D.

7.2. Contribuições do Trabalho

Esse projeto de doutorado teve como objetivo utilizar algoritmos genéticos e redes neurais como processos de otimização na predição de estruturas secundárias de peptídeos e proteínas, bem como prever a estrutura terciária de núcleos hidrofóbicos no interior de proteínas. Dois tipos de bibliotecas são investigados: dependentes ou não da estrutura da cadeia principal da proteína. Inicialmente a interação entre os átomos consiste basicamente do termo van der Waals.

O Trabalho possuiu ainda por objetivo investigar explorar como as características das proteínas que constituem as bases de dados para treinar as redes neurais podem influenciar no desempenho da performance das mesmas para diferentes arquiteturas de Redes Neurais.

Portanto, como contribuições desse trabalho pode-se mencionar:

- A investigação do uso de dois tipos de bibliotecas de rotâmeros na predição da estrutura terciária e no *design* de seqüências de núcleo hidrofóbicos e peptídeos. Permitindo observar que o uso desse tipo de biblioteca, restringindo o espaço conformacional, não prejudica, de maneira considerável, a predição de núcleos hidrofóbicos e peptídeos.
- O estudo de como diferentes bases de dados (toda- α , toda - β , α/β e mistas) podem auxiliar e melhorar a performance de redes neurais artificiais.



- Que a inclusão de potenciais de solvatação implícitos em algoritmos genéticos para predição 3D de peptídeos melhora de forma considerável a predição de peptídeos.

- A observação que a arquitetura da rede e principalmente a codificação problema são fatores limitantes na performance da predição 1D através de redes neurais do tipo MLP. E a observação que o fato de incluir mais informações para a rede MLP não implica em um acréscimo na performance da mesma.

- O estudo de que o uso de bibliotecas dependentes do backbone pode ser eficiente para a predição da estrutura de peptídeos quando o backbone é relaxado.

- O uso excessivo ou a codificação de informações físico-químicas pode prejudicar o desempenho das redes neurais. Porém o uso apropriado dessas propriedades pode melhorar o desempenho delas.

Como parte dos resultados do trabalho, foi projetado e está sendo desenvolvido um sistema de predição 3D (chamado YODA) e outro para predição 2D (baseado em redes neurais).



Referências Bibliográficas

7.3. Trabalhos Futuros

Como trabalhos que estão sendo realizados com o objetivo de conclusão do projeto de doutorado, pode-se citar:

- Aperfeiçoamento do sistema de predição 1D desenvolvido com a adição de júri, informações evolutivas, novas bases de dados.
- Testes de novas arquiteturas de redes neurais
- A predição e da estrutura e design de outros núcleos hidrofóbico utilizando AG.
- Teste de novas bibliotecas de rotâmeros na predição e design de núcleo hidrofóbico usando AG.
- Predição da estrutura e design do núcleo de mais peptídeos usando AG.
- Aperfeiçoamento do campo de força do Algoritmo Genético com a inclusão de outros termos de energia: um termo para formação de pontes de hidrogênio.
- O uso de algoritmos genéticos em conjunto com as redes neurais.

Referências Bibliográficas

BALDI, P. **Exploiting the past and the future in protein secondary structure prediction.** *Bioinformatics*, v. 15:11, November, p. 937-946, 1999.

BARTON, G. **Protein secondary structure prediction.** *Current Opinion in Structural Biology*, n. 5, p. 372-276, 1995.

BAUCOM, A.; CLINE, M.; HAUSSLER, D.; GREGORET, L. M. **Prediction of beta-sheet structure using neural networks.** *10th Annual Protein Society Meeting*, San Jose, California, August 3-7, 1996.

BOHR, H.; BOHR, J.; BRUNAK, S.; COTTERILL, R. M. J.; LAUTRUP B.; NOSKOV, L.; OLSEN, O.; PETERSEN, S. **Protein secondary structure and homology by neural networks.** *FEBS Letters*, N. 241, p. 223-228, 1998.

BOHM, G. **Review: New approaches in molecular structure prediction.** *Biophysical Chemistry*, n. 59, p. 1-32, 1996.

BRAGA, A. P.; CARVALHO, A. C. P. L.; LUDEMIR, B. T. **Fundamentos de redes neurais artificiais.** Recife- XI Escola Brasileira de Computação. SBC, 1998.

CAI Y., LI, Y.; CHOU, K. **Using Neural Networks for prediction of domain structure.** *Biochimica et Biophysica Acta*, n. 1476, p. 1-2, 2000.

CAI Y., LI, Y.; CHOU, K. **Classification and prediction of β - turns types by neural networks,** *Advances in Engineering Software*, n. 30, p. 347-352, 1999.

CHANDONIA, J. M.; KARPLUS, M. **New methods for accurate prediction of protein secondary structure ,** *PROTEINS: Structure, Function and Genetics*, n. 35, p. 293-306, 1999.



CHANDONIA, J. M.; KARPLUS, M. **The Importance of larger data sets for protein secondary structure prediction with neural networks**, *Protein Science*, n. 5, p. 768-774, 1996.

COMPIANI, M. **An entropy criterion to detect minimally frustrated intermediates in native proteins**, *Proc. Nat. Acad. Sci. USA*, v. 95, pp. 9290-9294, 1998.

CHAN, S. H.; DILL, K. **The Protein folding problem**, *Physics Today*, p. 24-32, 1993.

CREIGHTON, T. E. **Proteins: Structures and Molecular Properties**. Second Edition, Freeman, 1997.

CRIVELI, S.; BYRD, R.; ESKOU, E.; SCHNABE, R.; HEAD-GORDON T. **A Global optimization strategy for predicting α -helical protein tertiary structure**, *Computer Chemistry*, n. 24, p. 489-497, 2000.

CUFF, J. A.; BARTON, J. G. **Application of multiple sequence alignment profiles to improve protein secondary structure predictor**, *PROTEINS: Structure, Function and Genetics*, n. 40, p. 502-511, 2000.

CUFF, J. A.; BARTON, J. G. **Evaluation and improvement of multiple sequence methods for protein secondary structure prediction**, *PROTEINS: Structure, Function and Genetics*, n. 34, pp. 508-519, 1999.

CUFF, A. J.; CLAMP, E. M.; FINLAY M.; BARTON, J. G. **Jpred: a consensus secondary structure prediction server**, *Bioinformatics Applications Note*, v. 14, n. 10, 1998.

CUI Y. **Protein folding simulation with genetic algorithm and supersecondary structure constraints**, *Proteins: Structure, Functions and Genetics*, n. 31, pp. 247-257, 1998.

DANDEKAR, T.; ARGOS, P. **Folding the main chain of small proteins with the genetic algorithm**, *Journal of Molecular Biology*, n. 236, p. 844-861, 1998.



DANDEKAR, T.; ARGOS, P. **Potential of genetic algorithms in protein folding and protein engineering simulations**, *Protein Engineering*, n. 5, p. 637-645, 1992.

DESJARLAIS, J. R.; HANDEL, T. M. **Side-chain backbone flexibility in protein core design**, *Journal of Molecular Biology*, v. 290, Issue I, July, p. 305-318, 1999.

DILL, K. A. **Principles of protein folding – a perspective from simple exact models**. *Protein Science*, n. 4, p. 561-602, 1995.

DILL, K. A. **Dominate forces in protein folding**. *Biochemistry*, n. 29, 1990.

DINNER, R. A. et al. **Use of quantitative structure-property relationships to predict the folding ability of model proteins**. *PROTEINS: Structure, Functions and Genetics*, n. 33, p. 177-203, 1998.

DUNBRACK Jr., R. L. . and KARPLUS, M. **Backbone-dependent Rotamer Library for Proteins: Application to Side-chain prediction**. *J. Mol. Biol.*, **230**, 543-574, 1993.

DUNBRACK Jr, R. L. and KARPLUS, M. **Conformational analysis of the backbone-dependent rotamer preferences of protein sidechains**. *Nature Structural Biology*, **1**, 334-340, 1994.

FAUSSET, L. **Fundamentals of neural networks : architectures, algorithms, and applications**. New Jersey: Prentice Hall International Edition, 1994.

FERSHT, A. **Structure and mechanism in protein science**. New York: W. H. Freeman and Company Ed., 1998.

FETRON, J. S. et al **Patterns, and amino acid frequencies in structural building, a protein secondary structure classification scheme**. *Protein-Structure Function and Genetics*, vol. 27, n. 2 - February, p. 249-271, 1997.

FRIEDRICHS, M. S.; WOLYNES, P. G. **Genetic Algorithms for model bimolecular optimization problems**. (*Unpublished*), 1996.



GOLDBERG, E. D. **Genetic algorithms in search, optimization and machine learning**. Addison Wesley, 1989.

GOONATILAKE, G.; KHEBBAL, S. **Intelligent hybrid systems**. New York: John Wiley & Sons Inc., 1995.

HAYKIN, S. **Neural Networks: a comprehensive foundation**. IEEE Press, 1994.

HEAD-GORDON, T.; STILINGER H. F. **Optimal neural networks for protein-structure prediction**. *Physical Review E*, v. 48, n. 2, August p. 1502-1515, 1993.

HOLLEY, L. H.; KARPLUS M. **Neural networks for protein structure prediction**. *Methods in Enzymology*, v. 202, p. 204-224, 1991.

JUDSON, R. S. **Teaching polymers to fold**. *Journal Physic. Chemistry*, n. 96, p. 10102-10104, 1992.

KNELLER, G. D. **Improvements in protein secondary structure prediction by an Enhanced neural network**. *Journal of Molecular Biology*, p. 171-181, 1990.

KONO, H.; DOI J. **Energy minimization method using automata network for sequence and side-chain conformation prediction from given backbone geometry**. *Proteins: Structure, function and Genetics*, v. 19, p. 244-255, 1993.

KONIG, .; DANDEKAR, T. **Improving genetic algorithms for protein folding simulations by systematic crossover**. *Biosystems*, v. 50, pp. 17-25, 1999.

KIM, P. S.; BALDWIN, R. L. **Specific intermediates in the folding reactions of small proteins and the mechanism of protein folding**. *Annu. Rev Biochemistry*, n. 18, p. 459-489.

KOLLMAN, P. A.; DJAM Y. and LEE, M. R. **State of the art in studying protein Folding and protein structure prediction using MD Methods**. *J. Molecular Graphs and Modelling*, 2001, pp. 146-149.

KÓVACS, Z. L. **Redes neurais artificiais, fundamentos e aplicações.** São Paulo: Edição Acadêmica, 1996.

LESK, M. A. & Boswell R. D. **Homology modeling: Intreferences foorm Tables of Aligned Sequences.** Cur. Op. Structure Biology, 1992, 2:242-247.

LE GRAND, S. M.; MERZ, K. M. Jr. **The application of the genetic algorithm to the minimization of potential energy functions.** J. Global Optimization, n. 3, p.49-66, 1993.

LE GRAND, M. S.; Merz, K. M. Jr. **The Genetic algorithm and protein tertiary structure prediction.** *The Protein Folding Problem and Tertiary Structure Prediction*, p. 109-124, 1994.

MACLIN, R.; SHALVICK W. J. **Using knowledge-based neural Networks to improve algorithms: refining the chou-fasman algorithm for protein folding.** *Machine Learning*, n. 11, p. 195-215, 1993.

MCGUFFIN, L.; BRYSON, K.; Jones, T. D. **The PSIPRED protein structure prediction server.** *Bioinformatics Application Notes*, vol. 16, n.4, p. 892-893, 2000.

MATHEUS, K. C.; VAN HOLDE, E K. **Biochemistry.** The Benjamin/Cummings Publishing Company Inc., 1990.

MCCULLOCK, W. S.; PITTS W. H. **A Logical calculus of Ideas immanent in Nervous Activity.** *Bull Math Biophysics*, p. 115-133, 1943.

MURRAY D. **Tuning neural networks with genetic algorithms.** *AI Expert*, n. 9, p. 27-31, 1994.

MUSKAL, M. S.; KIM, H. S. **Predicting protein secondary structure content a tandem neural network approach.** *Journal of Molecular Biology*, v. 225, p. 713-727, 1992.

PENDERSEN, T. J.; Moult, S. J. **Genetic Algorithms for Protein Structure Prediction,** Cur. Op. Structure Biology, 6, 1996, pp. 227-231.



PENDERSEN, T. J.; Moult, J. **Protein folding simulations with genetic algorithms and a detailed molecular description**, *Journal of Molecular Biology*, v. 269, p. 240-259, 1997.

PONDER, J. W.; RICHARDS F. M. **Tertiary templates for protein use packing criteria in the enumeration of allowed sequences for different structural classes**. *J. Molecular Biology*, v. 193, p. 775-791, 1987.

QIAN, H. and SEJNOWSKI **Prediction of α -Helices in proteins based on thermodynamic parameters from solution chemistry**. *Journal of Molecular Biology*, n. 256, p. 663-666, 1996.

RABOW A. A. and SCHERAGA A. H. **Lattice neural network minimization: application of neural network optimization for locating the global-minimum conformations of proteins**. *Journal of Molecular Biology*, v. 232, p. 1157-1168, 1993.

ROMERO, P. **Intelligent data analysis for protein disorder prediction**. *Artificial Intelligence Review*, v. 14, n. 6, p. 447-484, 2000.

ROST, B.;Schneider R. e Sander, C. **Progress in Protein Structure Prediction?** TIBS 18, April 1993, pp. 120-123.

ROST, B.; SANDER, C. **Prediction of protein secondary structure at better than 70% accuracy**. *Journal of Molecular Biology*, v. 232, p. 584-599, 1993.

ROST B.; SANDER C. **Combining evolutionary information and neural networks to predict protein secondary structure**. *PROTEINS: Structure, Functions and Genetics*, v. 19, pp. 55-72, 1994.

ROST, B.; SANDER C. **Third Generation prediction of secondary structure**. *Protein structure prediction: Methods and Protocols*, p.71-95, 1999.

ROST, B. **Protein secondary structure prediction continues to rise**. *Current Opinion in Structural Biology* (pending) .



ROST, B. **Evolution teaches neural networks to predict protein structure.** *Scientific applications of neural nets'*, p. 207-223, 1999.

ROST, B. **Protein structure prediction in 1D, 2D and 3D.** *The Encyclopedia of Computacional Chemistry*, n. 3, p. 2242-2255, 1998.

ROST B. **PHD: predicting one-dimensional protein structure by profile based neural network.** *Methods in Enzymology*, n. 266, p. 525-539, 1996.

ROST B. **Better 1D predictions by experts with machines.** *Proteins*, Suppl. 1, p. 192-197, 1997.

ROSENBLUTH, A.; ROSENBLUTH, A.;TELLER, A.H; TELLER, E.J. **Metropolis.** *J. Chem. Phys.*, 21, 1087, 1953.

SAMUDRALA, R.; HUANG, S. E.; KOEHL, P.; LEVITT, M. **Constructing side chains on near-native main chains for *ab initio* protein structure prediction.** *Protein Engineering*, V. 13, n. 7, p. 453-457, 1997.

SKOLNICK, J. and KOINSKI, A. **Simulation of the folding protein.** *Science*, n. 250, p. 1121-1125, 1990.

STRYER, L. **Bioquímica.** Rio de Janeiro:Guanabara Koogan, 4ª. Edição, 1995.

SUN, S. **A Genetic algorithm that seeks native states of peptides and proteins.** *Biophysical Journal*, vol 69, August, pp. 340-355, 1995.

UNGER, R.; MOULT, J. **Genetic algorithms for protein folding simulations,** *Journal of Molecular Biology*, V. 231, p. 75-81, 1993.

VASCONCELOS, A. T. **Bioinformática: Análise de Banco de Dados Genéticos**, II Escola de Verão: Métodos Computacionais em Biologia, pp. 47-55, 2001.



VOET, D. e VOET, J. **Biochemistry**. Second Edition, Wiley, pp. 202-205, 1995.

WANG, J., WANG W., HUO, S. LEE M. and KOLLMAN, A. **Solvation Model Based on Weighted Solvent Accessible Surface Area**, J. Phys. Chem. B, 2001, 105, pp. 5055-507.

BARTLOW, M. Unsupervised Learning. *Neural Computation*, v. 1, p. 295-311, 1989.

BIDJALA, B. HICSSER, J., WAMER, R. Improving genetic algorithms with chemical constraints for proteins structure prediction. *Protein Protein Solving for Nature 2004*, n. 142, p. 232-267, 2004.

BRENNER, D. G. *Neural networks for pattern recognition*. Oxford University Press, 1993.

FRISMAN, D., ARGOB, H. Identification of tyrosine interactions in protein secondary structure predicted from amino acid sequence. *Protein Engineering*, vol. 9, n.2, p. 133-142, 1996.

GALLANT, A. *Neural network learning and expert systems*. MIT Press, 1993.

JACOBIANI, L. MARTELLI, P. L. FANSELLI, P., De Pisis, V., Casadei, R. Prediction of the transmembrane regions of beta-barrel membrane proteins with a neural network based predictor. *Protein Science*, vol. 10, n. 4, April, p. 779-787, 2001.

LELUX, J. A New algorithm for analysis of homology in protein primary structure. *Computational Chemistry*, v. 22, n. 1, p. 123-131, 2007.

LUNAS, R. R. *The Biology of Cells - from currents to networks*. Wile, Freeman and Company Ed., 1997.

SNOW, M. E. Parallel simulated-annealing algorithm locates global minimum of protein-folding potentials from multiple starting conformations. *J Comp. Chem.*, v. 13, pp. 289-297, 1992.

Bibliografia Consultada

ANNET, F. **Inflection points and chaotic behavior in searching the conformational space of cyclononane.** *J. Am. Chem. Society*, n. 112, p. 7172-7178, 1990.

BARLOW, H. **Unsupervised Learning.** *Neural Computation*, n. 1, p. 295-311, 1989.

BIDEALD, E.; HESSER, J.; MANER, R., **Implementing genetic algorithms with sterical constraints for proteins structures prediction.** *Parallel Problem Solving from Nature- PPSNV*, n. 1498, p. 959-967, 1998.

BISHOP, C. M. **Neural networks for pattern recognition.** Oxford University Press, 1996.

FRISHMAN, D.; ARGOS, P. **Incorporation of non-local interactions in protein secondary structure prediction from amino acid sequence.** *Protein Engineering*, vol. 9, n.2, p. 133-142, 1996.

GALLANT, S. **Neural network learning and expert systems.** MIT Press, 1993.

JACOBANI, I.; MARTELLI P. L.; FANISELLI, P.; De Pinto, V.; Casado, R. **Prediction of the transmembrane regions of beta barrel membrane proteins with a neural networks based predictor.** *Protein Science*, vol. 10, n. 4, April, p. 779-787, 2000.

LELUK, J. **A New algorithm for analysis of homology in protein primary structure.** *Computers Chemistry*, v. 22., n. 1, p. 123-131, 2000.

LLINÁS, R. R. **The Biology of brain – from neurons to networks.** WH. Freeman and Company Ed., 1989.

SNOW, M. E. **Powerful simulated-annealing algorithm locates global minimum of protein-foldig potentials from multiple starting conformations.** *J. Comp. Chem.*, v. 13, pp. 584-597, 1992.



STANDLEY, D. M. et. al **Tertiary structure prediction of mixed α/β proteins via energy minimization.** *Proteins: Structure, Function and Genetics*, v. 32, Issue 2, pp. 240-252, 1998.

TUFFERY, P.; ETCACHEBEST, C.; HAZOUT, S.; LAVER, R. **A new approach to the rapid determination of protein side chain conformations.** *J. Biomol. Struct. Dynam.*, V. 8,N. 6, p. 1267-1289, 1991.



Anexo 1

Exemplo de arquivo DSSP

Os DSSP são arquivos de caracteres ASCII que são construídos através de arquivos PDB e do software DSSP.

**** SECONDARY STRUCTURE DEFINITION BY THE PROGRAM DSSP, UPDATED CMBI VERSION BY ELMK /
APRIL 1,2000 **** DATE=24-SEP-2002
REFERENCE W. KABSCH AND C.SANDER, BIOPOLYMERS 22 (1983) 2577-2637

HEADER ELECTRON TRANSPORT 21-AUG-96 1JER
 COMPND 2 MOLECULE: CUCUMBER STELLACYANIN;
 SOURCE 2 ORGANISM_SCIENTIFIC: CUCUMIS SATIVUS;
 AUTHOR P.J.HART,A.M.NERSISSIAN,R.G.HERRMANN,R.M.NALBANDYAN,

```

T0 T1 I1 O1 TOTAL NUMBER OF RESIDUES, NUMBER OF CHAINS, NUMBER OF SS-  
BRIDGES(TOTAL,INTRACHAIN,INTERCHAIN)  
6218.0 ACCESSIBLE SURFACE OF PROTEIN (ANGSTROM**2)  
68 61.8 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(J) , SAME NUMBER PER 100  
RESIDUES  
7 6.4 TOTAL NUMBER OF HYDROGEN BONDS IN PARALLEL BRIDGES, SAME NUMBER PER 100  
RESIDUES  
16 14.5 TOTAL NUMBER OF HYDROGEN BONDS IN ANTIPARALLEL BRIDGES, SAME NUMBER PER 100  
RESIDUES  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I-5), SAME NUMBER PER 100 RESIDUES  
  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I-4), SAME NUMBER PER 100 RESIDUES  
  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I-3), SAME NUMBER PER 100 RESIDUES  
  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I-2), SAME NUMBER PER 100 RESIDUES  
  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I-1), SAME NUMBER PER 100 RESIDUES  
  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I+0), SAME NUMBER PER 100  
RESIDUES  
0 0.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I+1), SAME NUMBER PER 100  
RESIDUES  
16 14.5 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I+2), SAME NUMBER PER 100  
RESIDUES  
13 11.8 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I+3), SAME NUMBER PER 100  
RESIDUES  
11 10.0 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I+4), SAME NUMBER PER 100  
RESIDUES  
2 1.8 TOTAL NUMBER OF HYDROGEN BONDS OF TYPE O(I)->H-N(I+5), SAME NUMBER PER 100  
RESIDUES  
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 *** HISTOGRAMS OF ***  
  
0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 RESIDUES PER ALPHA HELIX  
0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 PARALLEL BRIDGES PER LADDER  
0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ANTIPARALLEL BRIDGES PER  
LADDER  
0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 LADDERS PER SHEET  
# RESIDUE AA STRUCTURE BP1 BP2 ACC N-H->O O->H-N N-H->O O->H-N TCO KAPPA ALPHA PHI  
PSI X-CA Y-CA Z-CA  
1 0 M - 0 0 127 0,0 2,-0.1 0,0 0,0 0.000 360.0 360.0 360.0 112.2 45.5 24.6 65.7  
2 1 Q - 0 0 186 1,-0.1 2,-0.1 34,-0.0 34,-0.1 -0.516 360.0-119.3 -73.7 148.0 45.2 25.1 61.9  
3 2 S - 0 0 60 32,-0.2 2,-0.4 -2,-0.1 33,-0.1 -0.212 35.8-103.6 -73.2 166.0 41.7 25.5 60.4  
4 3 T - 0 0 73 29,-0.2 33,-2.3 -2,-0.1 2,-0.7 -0.822 15.2-149.6-103.1 145.6 40.6 23.0 57.8  
5 4 V E -a 37 OA 79 -2,-0.4 2,-0.4 31,-0.2 33,-0.2 -0.904 24.2-154.5-106.1 111.3 40.6 23.5 54.0  
6 5 H E -a 38 OA 13 31,-3.2 33,-3.0 -2,-0.7 2,-0.7 -0.767 8.8-136.7 -88.7 140.8 37.7 21.2 52.9  
7 6 I E >-a 39 OA 51 -2,-0.4 3,-2.7 31,-0.2 33,-0.2 -0.839 26.6-123.8 -97.9 109.7 37.8 19.9 49.3  
8 7 V T 3 S+ 0 0 0 31,-2.4 33,-0.4 -2,-0.7 5,-0.2 -0.367 94.3 1.8-58.4 123.6 34.2 20.2 48.0  
9 8 G T >-S- 0 0 3 3,-0.1 3,-2.2 14,-0.1 -1,-0.3 0.569 102.2-124.1 75.2 10.4 33.1 16.8 46.9  
10 9 D G X S- 0 0 73 -3,-2.7 3,-0.9 1,-0.3 4,-0.0 -0.264 85.9 -2.7 48.9-128.0 36.4 15.2 48.0  
11 10 N G 3 S+ 0 0 175 1,-0.3 1,-0.3 2,-0.1 -2,-0.1 0.708 132.6 63.6-67.3-19.3 38.0 13.4 45.0  
12 11 T G <-S- 0 0 84 -3,-2.2 -1,-0.3 1,-0.2 -2,-0.2 0.831 87.1-167.3 -75.0-30.8 35.1 14.2 42.8  
13 12 G <- 0 0 5 -3,-0.9 2,-0.7 -4,-0.3 -1,-0.2 -0.389 42.4-34.3 76.5-164.0 35.6 17.9 42.9  
14 13 W S S+ 0 0 2 27,-2.8 2,-0.3 -2,-0.1 86,-0.1 -0.884 86.9 116.0-104.2 108.1 33.0 20.5 41.7  
15 14 S S S- 0 0 38 -2,-0.7 75,-0.0 27,-0.1 -2,-0.0 -0.917 75.2-67.2-154.8-179.7 31.0 19.3 38.7  
16 15 V - 0 0 56 -2,-0.3 83,-0.1 84,-0.1 84,-0.1 -0.707 62.2-135.5-80.1 122.7 27.4 18.4 37.8  
17 16 P - 0 0 18 0,0 83,-0.1 0,0 -1,-0.0 -0.307 11.8-129.6-82.9 161.6 26.7 15.2 39.8  
18 17 S S S+ 0 0 129 1,-0.2 -2,-0.0 81,-0.1 3,-0.0 0.872 102.1 42.9-74.4-35.6 25.0 12.0 38.7
```


19	18	S S> S-	0 0 64	1,-0.1	3,-2.0	80,-0.1	4,-0.2	-0.916	81.1-138.4-112.1	135.7	22.7	12.2	41.7		
20	19	P T3 S+	0 0 102	0,0.0	3,-0.3	0,0.0	4,-0.2	0.665	101.7	57.8	-62.9	-16.5	21.2	15.5	42.7
21	20	N T3> S+	0 0 57	1,-0.2	4,-2.3	2,-0.1	5,-0.3	0.350	75.4	107.6	-94.2	4.4	21.8	14.9	46.4
22	21	F H<> S+	0 0 61	-3,-2.0	4,-1.7	1,-0.2	-1,-0.2	0.908	80.0	38.9	-50.6	-53.3	25.5	14.4	46.0
23	22	Y H> S+	0 0 2	76,-0.4	4,-2.5	-3,-0.3	-1,-0.2	0.836	113.0	55.4	-78.0	-23.1	26.8	17.7	47.5
24	23	S H> S+	0 0 69	-4,-0.2	4,-2.0	1,-0.2	-1,-0.2	0.887	109.2	47.9	-78.1	-26.4	24.2	17.7	50.3
25	24	Q H X S+	0 0 132	-4,-2.3	4,-0.7	2,-0.2	-1,-0.2	0.908	111.9	51.1	-71.2	-34.4	25.3	14.3	51.4
26	25	W H>< S+	0 0 20	-4,-1.7	3,-1.1	-5,-0.3	-2,-0.2	0.963	111.2	47.0	-64.7	-45.6	29.0	15.4	51.2
27	26	A H>< S+	0 0 11	-4,-2.5	3,-1.8	1,-0.3	-1,-0.2	0.886	104.8	60.6	-65.1	-35.7	28.3	18.5	53.4
28	27	A H3< S+	0 0 89	-4,-2.0	-1,-0.3	1,-0.3	-2,-0.2	0.659	98.5	59.1	-70.1	-12.8	26.3	16.5	56.0
29	28	G T<< S+	0 0 65	-3,-1.1	2,-0.3	-4,-0.7	-1,-0.3	0.424	103.1	61.1	-89.3	0.5	29.4	14.4	56.7
30	29	K < -	0 0 71	-3,-1.8	2,-0.4	-4,-0.1	-24,-0.0	-0.838	69.4	136.9	-122.1	166.5	31.6	17.4	57.7
31	30	T -	0 0 93	-2,-0.3	2,-0.5	73,-0.0	75,-0.1	-0.991	22.1	-156.4	-121.1	127.0	31.6	20.1	60.4
32	31	F E -c	106 0B 7	73,-0.7	75,-2.9	-2,-0.4	2,-0.3	-0.945	10.5	-170.9	-108.8	127.9	32.5	23.6	59.1
33	32	R E> -c	107 0B 108	-2,-0.5	3,-2.8	73,-0.2	75,-0.2	-0.881	36.8	-89.7	-119.4	150.8	33.9	26.2	61.5
34	33	V T3 S+	0 0 46	73,-2.5	47,-0.2	-2,-0.3	3,-0.1	-0.205	117.4	36.3	-52.2	135.4	34.6	29.9	61.0
35	34	G T3 S+	0 0 8	45,-3.3	-1,-0.3	1,-0.4	-32,-0.2	0.112	92.1	111.2	103.4	-26.8	38.2	30.2	59.7
36	35	D < -	0 0 7	-3,-2.8	44,-3.0	44,-0.3	-1,-0.4	-0.385	56.8	-143.7	-76.6	170.5	38.0	27.0	57.5
37	36	S E -aB	5 79A 18	-33,-2.3	-31,-3.2	42,-0.2	2,-0.5	-0.890	4.4	-135.9	-131.2	163.8	38.0	27.4	53.7
38	37	L E -aB	6 78A 0	40,-2.7	40,-3.1	-2,-0.3	2,-0.5	-0.988	15.5	-156.7	-120.0	127.0	36.2	25.5	50.9
39	38	Q E -aB	7 77A 44	-33,-3.0	-31,-2.4	-2,-0.5	2,-0.6	-0.951	3.7	-164.2	-106.5	122.8	38.3	24.5	47.9
40	39	F E -B	0 76A 0	36,-3.3	36,-2.2	-2,-0.5	2,-0.5	-0.941	7.4	-169.8	-108.1	119.6	36.3	24.0	44.6
41	40	N +	0 0 56	-2,-0.6	-27,-2.8	-33,-0.4	33,-0.0	-0.939	21.9	140.1	-111.0	128.2	38.4	22.1	41.9
42	41	F -	0 0 16	-2,-0.5	2,-0.1	-29,-0.2	-27,-0.1	-0.982	59.0	-76.0	-157.9	164.3	37.0	21.9	38.4
43	42	P >> -	0 0 87	0,0.0	3,-2.0	0,0.0	4,-0.7	-0.389	64.0	-98.8	-68.0	130.4	38.3	22.1	34.8
44	43	A T34 S+	0 0 67	1,-0.3	29,-0.2	2,-0.1	4,-0.1	-0.073	103.3	12.5	-46.9	149.8	39.0	25.7	33.9
45	44	N T34 S+	0 0 120	27,-2.3	-1,-0.3	2,-0.2	28,-0.1	0.346	106.0	89.5	67.8	-10.4	36.3	27.4	32.0
46	45	A T<4 S+	0 0 40	3,-2.0	2,-0.3	26,-0.2	-1,-0.2	0.741	96.9	0.7	-91.2	-28.0	33.7	24.7	32.5
47	46	H < -	0 0 19	-4,-0.7	2,-0.3	25,-0.3	-2,-0.2	-0.869	61.2	-153.6	-142.0	174.4	32.0	25.9	35.7
48	47	N -	0 0 0	-2,-0.3	2,-0.4	-4,-0.1	44,-0.1	-0.874	16.6	-121.8	-152.4	179.5	32.2	28.7	38.3
49	48	V +	0 0 12	22,-0.4	2,-0.4	-2,-0.3	41,-0.2	-0.988	28.6	171.7	-135.1	123.3	31.5	29.3	41.9
50	49	H E -D	89 0B 40	39,-2.4	39,-2.6	-2,-0.4	2,-0.4	-0.995	24.8	-138.5	-133.0	133.2	29.1	32.0	43.1
51	50	E E -D	88 0B 87	-2,-0.4	2,-0.3	37,-0.2	37,-0.3	-0.726	20.5	-149.5	-89.0	136.5	27.8	32.6	46.7
52	51	M E -D	87 0B 5	35,-2.9	35,-0.6	-2,-0.4	11,-0.0	-0.854	9.2	-142.2	-104.0	155.5	24.1	33.4	47.0
53	52	E S S+	0 0 94	-2,-0.3	2,-0.3	33,-0.1	-1,-0.1	0.865	76.9	33.3	-77.5	-48.2	22.5	35.6	49.6
54	53	T > -	0 0 52	1,-0.1	4,-1.9	32,-0.0	5,-0.2	-0.858	68.4	-127.5	-121.0	154.5	19.2	33.9	50.3
55	54	K H> S+	0 0 92	-2,-0.3	4,-2.9	2,-0.2	5,-0.2	0.752	108.2	60.3	-64.2	-34.1	17.8	30.3	50.4
56	55	Q H> S+	0 0 139	2,-0.2	4,-2.2	1,-0.2	-1,-0.2	0.960	108.1	42.6	-64.0	-49.0	15.0	31.3	48.1
57	56	S H>> S+	0 0 18	1,-0.2	5,-2.4	2,-0.2	4,-0.6	0.890	115.9	49.2	-62.4	-44.6	17.3	32.3	45.3
58	57	F H>< S+	0 0 12	-4,-1.9	3,-1.0	2,-0.2	-1,-0.2	0.928	111.7	49.9	-59.1	-47.0	19.6	29.3	45.9
59	58	D H3< S+	0 0 74	-4,-2.9	-2,-0.2	1,-0.3	-1,-0.2	0.914	117.3	39.0	-61.6	-38.4	16.6	27.0	45.8
60	59	A H3< S-	0 0 68	-4,-2.2	-1,-0.3	-5,-0.2	-2,-0.2	0.484	103.3	-127.4	-89.3	-1.1	15.3	28.5	42.6
61	60	a T<< S+	0 0 18	-3,-1.0	2,-0.7	-4,-0.6	-3,-0.2	0.912	51.9	162.7	51.6	45.8	18.7	28.8	40.9
62	61	N <+	0 0 78	-5,-2.4	3,-0.3	-6,-0.1	-1,-0.2	-0.856	19.2	179.2	-101.2	113.7	17.7	32.4	40.3
63	62	F > +	0 0 54	-2,-0.7	3,-2.7	1,-0.2	-1,-0.1	0.111	38.1	123.7	-105.9	22.9	20.8	34.5	39.6
64	63	V T3 S+	0 0 118	1,-0.3	-1,-0.2	3,-0.1	-2,-0.0	0.878	84.0	40.9	-56.2	-31.6	19.3	37.9	39.0
65	64	N T3 S+	0 0 111	-3,-0.3	-1,-0.3	2,-0.1	2,-0.1	0.394	98.4	115.0	-96.7	16.5	21.5	39.4	41.7
66	65	S S< S-	0 0 26	-3,-2.7	2,-1.0	1,-0.1	-3,-0.1	-0.411	74.4	-99.0	-86.4	156.7	24.5	37.3	40.5
67	66	D -	0 0 96	1,-0.1	2,-9.9	-2,-0.1	-1,-0.1	-0.732	37.4	-109.2	-95.6	103.3	27.6	38.6	39.0
68	67	N -	0 0 125	-2,-1.0	2,-0.6	2,-0.1	-1,-0.1	0.310	56.0	-114.2	7.7	-12.0	27.8	38.4	35.3
69	68	D +	0 0 38	-2,-9.9	2,-0.7	1,-0.0	-2,-0.0	-0.883	54.8	157.2	101.2	-125.9	30.3	35.7	35.7
70	69	V +	0 0 121	-2,-0.6	2,-0.6	2,-0.0	-2,-0.1	-0.541	53.9	67.8	103.8	-63.0	33.8	36.6	34.5
71	70	E +	0 0 122	-2,-0.7	-22,-0.4	1,-0.1	-2,-0.0	-0.846	39.0	158.7	-103.5	121.2	36.3	34.2	36.4
72	71	R +	0 0 75	-2,-0.6	-27,-2.3	-24,-0.1	-25,-0.3	-0.135	33.5	130.3	-123.5	27.4	36.2	30.5	35.7
73	72	T -	0 0 89	-29,-0.2	-2,-0.0	-28,-0.1	-29,-0.0	-0.633	51.8	-109.4	-87.1	150.9	39.7	29.7	36.9
74	73	S S S+	0 0 41	-2,-0.2	-33,-0.1	1,-0.1	2,-0.1	-0.938	85.2	46.3	-128.2	148.7	40.2	26.8	39.3
75	74	P -	0 0 67	0,0.0	2,-0.5	0,0.0	-34,-0.2	0.520	64.1	-173.1	-82.7	153.8	40.8	26.2	42.1
76	75	V E -B	40 0A 19	-36,-2.2	-36,-3.3	-2,-0.1	2,-0.5	-0.944	5.4	-166.6	-106.7	124.0	38.5	28.7	43.8
77	76	I E -B	39 0A 90	-2,-0.5	2,-0.5	-38,-0.2	-38,-0.2	-0.983	2.2	-165.7	-119.1	119.5	39.0	28.7	47.6
78	77	E E -B	38 0A 58	-40,-3.1	-40,-2.7	-2,-0.5	2,-0.3	-0.925	13.9	-136.0	-109.2	134.0	36.4	30.5	49.7
79	78	R E -B	37 0A 144	-2,-0.5	2,-1.0	-42,-0.2	-42,-0.2	-0.647	12.3	-134.6	-85.5	140.4	37.0	31.5	53.3
80	79	L +	0 0 0	-44,-3.0	-45,-3.3	-2,-0.3	-44,-0.3	-0.823	40.4	158.2	-100.3	99.9	34.2	30.8	55.8
81	80	D +	0 0 94	-2,-1.0	2,-0.3	-47,-0.2	-1,-0.2	0.470	51.1	49.3	-95.1	-23.5	34.1	34.1	57.7
82	81	E S S-	0 0 126	2,-0.1	24,-0.1	-49,-0.0	2,-0.1	-0.953	78.9	-110.4	-129.1	153.5	30.6	34.2	59.2
83	82	L S S+	0 0 112	-2,-0.3	24,-0.2	22,-0.1	2,-0.2	-0.347	73.4	34.0	-72.6	161.4	28.4	31.9	61.2
84	83	G E S-E	0 106B 25	22,-2.8	22,-2.6	-2,-0.1	2,-0.2	-0.527	97.3	-10.7	96.4	-169.8	25.3	30.4	59.7
85	84	M E -E	0 105B 89	20,-0.2	2,-0.3	-2,-0.2	20,-0.2	-0.496	52.4	-172.6	-74.7	138.6	24.1	29.1	56.4
86	85	H E -E	0 104B 15	18,-2.3	18,-2.4	-2,-0.2	2,-0.4	-0.960	6.0	-158.8	-130.8	144.4	26.1	29.7	53.2
87	86	Y E +DE	52 103B 31	-35,-0.6	-35,-2.9	-2,-0.3	2,-0.4	-0.987	10.6	178.8	-129.1	127.9	24.9	28.8	49.7
88	87	F E +DE	51 102B 1	14,-3.0	14,-2.8	-2,-0.4	2,-0.3	-0.975	6.2	171.1	-129.6	145.6	27.2	28.4	46.6
89	88	V E -DE	50 101B 0	-39,-2.6	-39,-2.4	-2,-0.4	2,-0.5	-0.915	36.0	-116.5	-142.5	169.6	26.2		



95	94	H T X4>S+	0	0	40	-3,-2.3	5,-2.5	1,-0.2	3,-0.5	0.912	76.5	49.2	-58.0	-42.8	24.2	23.1	36.2
96	95	a G >45S+	0	0	2	-6,-0.5	3,-2.2	-4,-0.4	-1,-0.2	0.971	110.6	49.7	-64.4	-45.0	22.5	24.6	39.4
97	96	S G 345S+	0	0	92	1,-0.3	-1,-0.3	-4,-0.2	-2,-0.2	0.764	107.8	57.0	-62.3	-22.4	19.2	23.1	38.5
98	97	N G <<5S-	0	0	87	-4,-1.4	-1,-0.3	-3,-0.5	-2,-0.2	0.206	132.1	-86.9	-92.0	11.7	21.0	19.7	37.9
99	98	G T < 5S+	0	0	7	-3,-2.2	2,-1.9	1,-0.2	-76,-0.4	0.582	84.2	133.2	103.3	1.8	22.4	19.7	41.4
100	99	Q < +	0	0	0	-5,-2.5	-10,-1.6	-6,-0.2	2,-0.3	-0.550	48.8	99.3	-89.0	77.3	25.6	21.6	41.2
101	100	K E - E	0	89B	46	-2,-1.9	2,-0.4	-12,-0.2	-12,-0.2	-0.969	52.6	-159.4	-157.0	152.0	24.9	23.7	44.3
102	101	L E - E	0	88B	2	-14,-2.8	-14,-3.0	-2,-0.3	2,-0.4	-0.999	5.0	-169.6	-144.1	130.7	26.0	23.4	48.0
103	102	S E + E	0	87B	37	-2,-0.4	2,-0.3	-16,-0.2	-16,-0.2	-0.995	15.5	178.1	-123.3	131.9	24.6	24.7	51.2
104	103	I E - E	0	86B	4	-18,-2.4	-18,-2.3	-2,-0.4	2,-0.6	-0.955	26.2	-133.2	-135.0	155.7	26.6	24.5	54.4
105	104	N E - E	0	85B	84	-2,-0.3	-73,-0.7	-20,-0.2	2,-0.5	-0.942	21.0	-155.2	-110.5	120.6	26.2	25.5	58.0
106	105	V E -cE	32	84B	2	-22,-2.6	-22,-2.8	-2,-0.6	-73,-0.2	-0.849	12.4	-172.3	-98.1	127.7	29.2	27.3	59.6
107	106	V E -c	33	0B	51	-75,-2.9	-73,-2.5	-2,-0.5	-76,-0.0	-0.654	32.6	-81.3	-112.8	170.8	29.7	27.1	63.4
108	107	A -	0	0	47	-2,-0.2	-1,-0.2	-75,-0.2	-26,-0.0	-0.171	27.2	-136.6	-66.1	168.7	32.1	28.9	65.6
109	108	A	0	0	60	-76,-0.1	-1,-0.1	1,-0.1	-76,-0.0	0.848	360.0	360.0	-94.9	-34.4	35.6	27.6	66.2
110	109	N	0	0	132	0, 0.0	-1,-0.1	0, 0.0	-2,-0.0	0.954	360.0	360.0	-67.7	360.0	35.9	28.0	69.9



