

UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”  
Programa de Pós-graduação em Ciência e Tecnologia de Materiais

PAULA MARTINS DA SILVA

**APLICAÇÕES DA TEORIA DE NÓS NO ESTUDO  
DA ESTRUTURA DE PROTEINAS**

**BAURU**

**2013**

Paula Martins da Silva

APLICAÇÕES DA TEORIA DE NÓS NO ESTUDO  
DA ESTRUTURA DE PROTEÍNAS

Tese apresentada como requisito à obtenção do título de Doutor à Universidade Estadual Paulista “Júlio de Mesquita Filho” – Programa de Pós-Graduação em Ciência e Tecnologia de Materiais, sob a orientação do Prof. Dr. Aginaldo Robinson de Souza.

BAURU

2013

## FICHA CATALOGRÁFICA

DA SILVA, Paula Martins

Aplicações da Teoria de Nós no estudo da Estrutura de Proteínas – Paula Martins da Silva, 2013. 129 f.

Orientador: Aguinaldo Robinson de Souza.

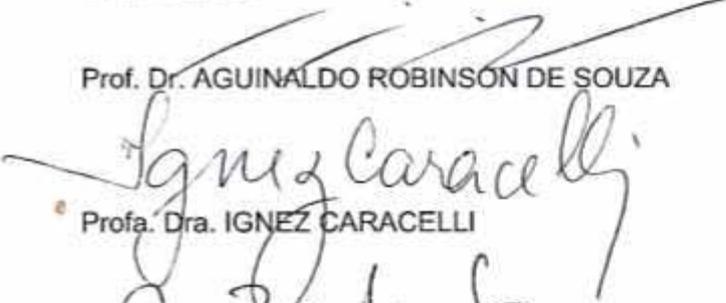
Tese (Doutorado) Universidade Estadual Paulista. Faculdade de Ciências, Bauru, 2013.

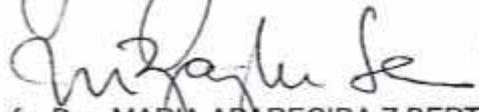
1. Simulação Computacional; 2. Teoria dos Nós; 3. Geometria da Proteína. I. Universidade Estadual Paulista. Faculdade de Ciências. II. Doutora em Ciência e Tecnologia de Materiais

**ATA DA DEFESA PÚBLICA DA TESE DE DOUTORADO DE PAULA MARTINS DA SILVA,  
DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA E TECNOLOGIA DE  
MATERIAIS, DO(A) FACULDADE DE CIÊNCIAS DE BAURU.**

Aos 20 dias do mês de novembro do ano de 2013, às 14:00 horas, no(a) Sala 1 da Pós-graduação / FC, reuniu-se a Comissão Examinadora da Defesa Pública, composta pelos seguintes membros: Prof. Dr. AGUINALDO ROBINSON DE SOUZA do(a) Departamento de Química / Faculdade de Ciências de Bauru, Profa. Dra. IGNEZ CARACELLI do(a) Departamento de Física / Universidade Federal de São Carlos, Profa. Dra. MARIA APARECIDA Z BERTOCHI do(a) Departamento de Bioquímica e Tecnologia Química / Instituto de Química de Araraquara, Prof. Dr. VALDECIR FARIAS XIMENES do(a) Departamento de Química / Faculdade de Ciências de Bauru, Prof. Dr. ANTONIO CALIRI do(a) Faculdade de Ciências Farmacêuticas / Universidade de São Paulo, sob a presidência do primeiro, a fim de proceder a arguição pública da TESE DE DOUTORADO de PAULA MARTINS DA SILVA, intitulada "APLICAÇÕES DA TEORIA DE NÓS NO ESTUDO DA ESTRUTURA DE PROTEÍNAS". Após a exposição, a discente foi arguida oralmente pelos membros da Comissão Examinadora, tendo recebido o conceito final: APROVADA. Nada mais havendo, foi lavrada a presente ata, que, após lida e aprovada, foi assinada pelos membros da Comissão Examinadora.

Prof. Dr. AGUINALDO ROBINSON DE SOUZA

  
Profa. Dra. IGNEZ CARACELLI

  
Profa. Dra. MARIA APARECIDA Z BERTOCHI

  
Prof. Dr. VALDECIR FARIAS XIMENES

  
Prof. Dr. ANTONIO CALIRI

“Se dois indivíduos estão em acordo em tudo, pode ter certeza que  
um dos dois pensa por ambos.”

*Sigmund Freud*

## **AGRADECIMENTOS**

A Deus, em primeiro lugar.

Aos meus alunos, coordenadores e funcionários da Universidade Paulista (UNIP) e ao Serviço Nacional de Aprendizagem Comercial (SENAC), ambos da cidade de Bauru.

Aos meus colegas e amigos da pós-graduação. Serei injusta caso cite nomes, pois, em minha trajetória nesses quatro anos, foram muitos os que contribuíram com a minha formação.

À minha família, pelo apoio, estímulo e carinho.

Ao meu orientador e ao Programa de Pós-Graduação em Ciência e Tecnologia de Materiais (POSMAT) da Universidade Estadual Paulista (UNESP), campus de Bauru.

SILVA, P. M. da. **Aplicações da teoria de nós no estudo da estrutura de proteínas**. 2013. 129 f. Tese (Doutora em Ciência e Tecnologia de Materiais) - UNESP, Faculdade de Ciências, Bauru, 2013.

## RESUMO

Este estudo consiste em entender a forma tridimensional de proteínas, investigando padrões de comportamento na área do Nó matemático em relação à proteína real. A abordagem adotada envolveu a aplicação de uma teoria matemática sobre os Nós (*Knots*) para o entendimento da geometria adotada pelas proteínas. No laboratório de simulação computacional da UNESP, campus de Bauru, foi utilizado um microcomputador Pentium com 1GB de RAM. Os *softwares* utilizados foram: Origin 7.0, pacote da Microsoft Office, CodeBlocks - que é um ambiente de desenvolvimento integrado (IDE) para plataformas Linux -, MAC OS e Windows, que permite escrever aplicativos para ambiente gráficos ou aplicativos de console em linguagem C ou C++, com suporte a múltiplos compiladores como GCC/MinGW, SDCC, Intel C++, GNU ARM. Além destes, o *software* RasMol foi utilizado para visualização das proteínas e o *software* KnotPlot, para investigação dos Nós matemáticos. O método apresenta o levantamento, tratamento e integração dos dados. Através do *software*, não foram reconhecidos padrões em proteínas que possuem os Nós. Apresentou-se uma nova abordagem para visualizar alfa hélice, fitas betas e turns. No que diz respeito às distâncias internas, foi possível analisar o comportamento das proteínas e, na topologia matemática a  $3_1$ , detectou-se similaridade em relação às quantidades de concavidades e picos. Concluiu-se que o programa necessita de parâmetros estabelecidos na literatura para realizar todas as leituras dos resultados apresentado pelo *software*.

**Palavras-chave:** Simulação Computacional; Teoria dos Nós; Geometria da Proteína.

SILVA, P. M. da. **Applications of the Knot theory in the study of proteins structure.** 2013. 129 f. - Thesis (Doctor in Science and Technology of Materials) - UNESP, Faculdade de Ciências, Bauru, 2013.

## **ABSTRACT**

The present study is aimed at understanding the three-dimensional form of proteins, thus investigating behavioral patterns of the mathematical Knot in relation to the actual protein. It was adopted the approach of applying a Math theory about Knots for the comprehension of the proteins geometry proteins. In the computer simulation laboratory at UNESP, Bauru campus, it was used a 1GB RAM PC Pentium, and the software Origin 7.0; Microsoft Office package; CodeBlocks, an integrated development environment for Linux platform; MAC OS and Windows, which allow to write applications for graphic environments or console applications in C or C++ language with support to multiple compilers such as GCC/MinGW, SDCC, Intel C++, and GNU ARM. Besides, the software RasMol was also used for the visualization of the proteins, and KnotPlot for the investigation of mathematical Knots. The method presents data gathering, processing and integration. By using the software there were no recognized patterns in proteins with Knots, and it was presented a new approach in order to visualize alpha-helix, beta-strands, and turns. In what concerns internal distances, it was possible to analyze the behavior of proteins and, in  $3_1$  Math topology, it was detected some similarity in relation to the amount of peaks and troughs. Thus, it was concluded that the program requires parameters established in the literature in order to perform all reading of the results presented by the software.

**Keywords:** Computer Simulation, Knot Theory, Protein Geometry

## LISTA DE FIGURAS

Figura 1 - Possível representação dos átomos como vórtices. ....	15
Figura 2 - Nó com número de ligações igual a quatro. ....	16
Figura 3 - Nó 31 no espaço cartesiano (x, y, z).....	16
Figura 4 - Distâncias internas no Nó 31. ....	17
Figura 5 - Topologias associadas a Nós (ADAM, 1994).....	21
Figura 6 - Topologia dos Nós Matemáticos (ADAM, 1994). ....	23
Figura 7 - Exemplo de conformações em malha quadrática (a) e malha cúbica (b) em Modelo HP. Esferas cinza ilustram os aminoácidos Hidrofóbicos e as brancas, os Polares. A malha 3x3x4 é apontada na (c). ....	24
Figura 8 - Representação Gráfica do Nó Trivial. ....	26
Figura 9 - Representação Gráfica do Trifólio.....	27
Figura 10 - Movimento Reidemeister 01 ....	29
Figura 11 - Movimento Reidemeister 02 ....	29
Figura 12 - Movimento Reidemeister 03 ....	29
Figura 13 - Representação gráfica dos cruzamentos.....	30
Figura 14 - Equações do sistema linear ....	31
Figura 15 - Orientação do diagrama K. ....	33
Figura 16 - Orientação definida. ....	34
Figura 17 - Coordenadas cartesianas para o cálculo de distância.....	35
Figura 18 - Carbonos alfa da proteína 2efv e suas distâncias internas em ångström (Å).....	36
Figura 19 - Ângulo entre dois planos.....	37
Figura 20 - Distância final-final.....	38
Figura 21 - Banco de Dados.....	39
Figura 22 - Crescimento anual do número total de estruturas 3D de proteínas no PDB. Dados acessados em novembro de 2013. O azul ilustra a quantidade de estruturas determinadas anualmente e o vermelho, o total.....	40
Figura 23 - Tela final do servidor. Nós por H. influenzae (PDB ID 1uam). ....	43
Figura 24 - Tela inicial do software executado. ....	45
Figura 25 - Tela inicial identificando a estrutura a ser executada.....	45
Figura 26 - Resultado executado corretamente. ....	46
Figura 27 - Arquivos que serão utilizados para cálculos. ....	46

Figura 28 - Menu Inicial.....	48
Figura 29 - Criação dos arquivos de qualquer opção.....	49
Figura 30 - Tela identificadora dos arquivos gerados corretamente.....	49
Figura 31 - Tela inicial após qualquer opção ter sido gerada.....	50
Figura 32 - Resultado do cálculo da Distância automática.....	51
Figura 33 - Distância interna entre C $\alpha$ da proteína 2efv. A cor preta representa a primeira posição do aminoácido carbono alfa. O azul indica o cálculo de distâncias a partir da segunda posição do aminoácido carbono alfa e o vermelho é a terceira posição do carbon .....	54
Figura 34 - Distância interna da proteína 2efv referente ao primeiro número do resíduo (em preto) e distância interna no plano do quadragésimo terceiro resíduo carbono alfa (em vermelho).....	55
Figura 35 - Distância interna da proteína 2efv referente ao primeiro número do resíduo (em preto) e cálculo do último resíduo C $\alpha$ (em vermelho).....	56
Figura 36 - Proteína 1ns5.....	58
Figura 37 - Distância interna da proteína 1ns5 cadeia A. Em vermelho, a demarcação da região do Nó. ....	59
Figura 38 - Distância interna do Nó Matemático 31 unidades arbitrárias .....	60
Figura 39 - Distância interna da proteína 2efv. Em vermelho, a demarcação da região do Nó.....	61
Figura 40 -Distâncias internas dos carbonos alfa da região do Nó, conforme literatura (VIRNAU, P.; MIRNY, A. L.; KARDAR, M., 2006).....	62
Figura 41 - Distâncias internas retirando cinco carbonos alfas de cada lado.....	63
Figura 42 - Proteína 1ztu.....	64
Figura 43 - Distância interna Proteina 1ztu. Demarcação em vermelho referente à topologia do Nó 41 .....	65
Figura 44 - Distância interna do Nó matemático 41 .....	66
Figura 45 - Proteína 2etl.....	67
Figura 46 - Distância interna topologia matemática 52 unidades arbitrárias .....	68
Figura 47 - Distância interna da cadeia A. Em vermelho, a região do Nó 52 .....	68
Figura 48 - Proteína 3bjx.....	69
Figura 49 - Distância interna da topologia matemática Nó 61 .....	70
Figura 50 -Distância interna proteína 3bjx. Em vermelho, região do Nó 61 .....	71
Figura 51 - Distâncias internas da Lisozima 2lyz .....	72

Figura 52 - PDBSum .....	73
Figura 53 - Diedros de todos os carbonos alfa. Em vermelho, Diedro da região do Nó.....	74
Figura 54 - Posição 1 Proteína 2efv .....	76
Figura 55 - Diedros da Lisozima.....	76
Figura 56 - Proteína 2efv e seus respectivos carbonos alfa indicando, em sua cadeia, a distância ponta-ponta.....	80
Figura 57 - Proteína 2efv e seus respectivos carbonos alfa indicando a distância ponta-ponta em sua configuração apenas da região do Nó.....	82
Figura 58 - Gráfico Hidrofobicidade em relação aos aminoácidos 2efv .....	85
Figura 59 - Frequência da Proteína 2efv região do Nó. ....	86
Figura 60 - Frequência de Aminoácidos.....	89
Figura 61 - Representação da equação paramétrica de uma hélice .....	92

## LISTA DE TABELAS

Tabela 1 - Informações das Proteínas utilizadas para analisar as distâncias internas .....	53
Tabela 2 - Planos para cálculos dos diedros.....	75
Tabela 3 - Mostra os cálculos dos diedros da proteína 2lyz.....	77
Tabela 4 - Distância Ponta-Ponta.....	79
Tabela 5 - Distância ponta-ponta – região do Nó .....	81
Tabela 6 - Escala Kyte-Doolittle .....	83
Tabela 7 - Hidrofobicidade da região do nó proteína 2efv.....	84
Tabela 8 - Frequência de Aminoácidos da Proteína 2efv – Região do Nó (Knots).....	86
Tabela 9 - Frequência de Aminoácidos da Proteína 1ns5.....	88

## SUMÁRIO

1. Introdução .....	14
1.1 Proteínas .....	17
1.2 Aminoácidos .....	18
1.3 Estrutura Primária .....	19
1.4 Estrutura Secundária .....	19
1.5 Estrutura Terciária .....	20
1.6 Estrutura Quaternária .....	20
1.7 Estudos sobre Nós (Knots) em Proteínas.....	21
1.8 Motivação e hipóteses de Pesquisa.....	23
1.9 Objetivos.....	24
2. Teoria dos Nós .....	25
2.1 Projeções.....	26
2.2 Movimentos de Reidemeister .....	28
2.3 Invariante Determinante.....	30
2.4 Invariante Polinômio Alexander .....	32
2.5 Distância Interna.....	35
2.6 Diedros .....	36
2.7 Distância Final – Final.....	38
2.8 Banco de Dados .....	39
3. Material e Métodos.....	41
3.1 Levantamentos de Dados .....	42
3.2 Tratamento .....	44
3.3 Integração dos Dados.....	50
4. Resultados e Discussão .....	52

4.1 Resultados: Distância Interna .....	52
4.2 Resultados: Diedros.....	73
4.3 Resultados Distância Final-Final proteína inteira.....	79
4.4 Resultados Distância Final-Final região do Nó .....	81
4.5 – Escala de Hidrofobicidade Região do Nó (Knot).....	83
5. Conclusões e considerações finais .....	90
6. Referências .....	93
7.ANEXOS .....	102

## 1 INTRODUÇÃO

O interesse científico inicial sobre Nós (*knots*) foi motivado pelos estudos do Químico Willian Thomson (Lord Kelvin) que, em 1868, apresentou a hipótese de que os átomos poderiam ser representados por estas entidades matemáticas no seio do éter, de modo que diferentes nós corresponderiam a elementos químicos específicos (THOMSON, 1868).

Na Figura 1, apresentam-se três tipos de Nós diferentes como uma possível representação dos elementos químicos Sódio (Na), Argônio (Ar) e Cobalto (Co) (ADAMS, 2004).

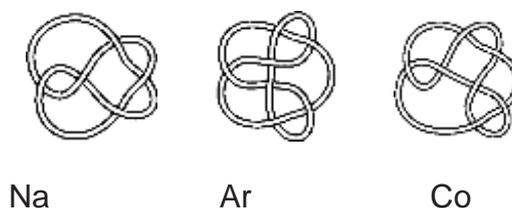


Figura 1 - Possível representação dos átomos como vórtices. Fonte: ADAMS, 2004.

No ano de 1887, com a confirmação experimental da não existência do éter, por Michelson e Morley, a hipótese atômica de Thomson foi abandonada pela comunidade científica.

Na Matemática, o estudo dos Nós obteve um significativo avanço com os estudos de Carl Friedrich Gauss que desenvolveu o conceito de número de ligação (*linking number*) como uma invariante numérica na

## FUNDAMENTAÇÃO TEÓRICA

---

descrição da ligação de duas curvas fechadas no espaço tridimensional. Este número representa o número de vezes que cada curva entrelaça a outra e é sempre um número inteiro, podendo ser positivo ou negativo a depender da orientação das duas curvas. Na Figura 2, apresentam-se duas curvas com o número de ligações igual a quatro (BAR-NATAN, 2013).

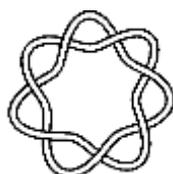


Figura 2 - Nó com número de ligações igual a quatro. Fonte: ADAMS, 2004.

Na Figura 3 ilustra o Nó  $3_1$  no espaço cartesiano (x, y, z).

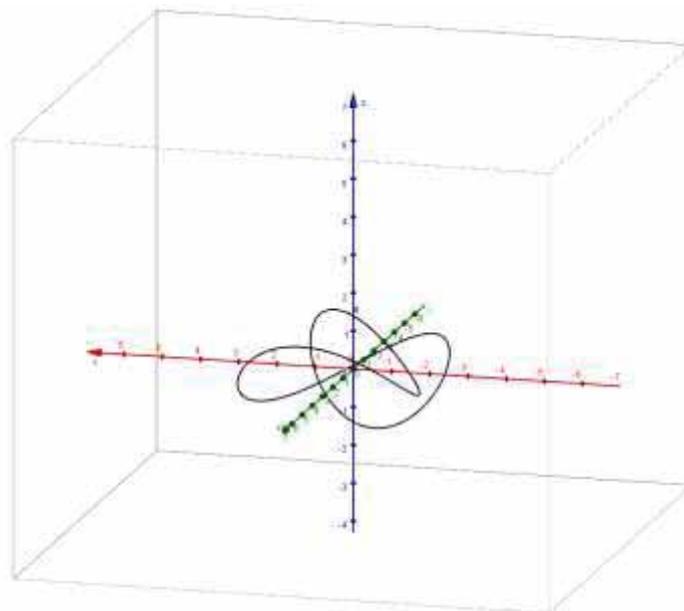


Figura 3 - Nó  $3_1$  no espaço cartesiano (x, y, z). Fonte: Laboratório Simulação Molecular.

## FUNDAMENTAÇÃO TEÓRICA

---

Uma característica interessante do Nó  $3_1$  é a sua simetria, que pode ser visualizada a partir do cálculo das distâncias internas dos pontos que representam os Nós. Na Figura 4, apresentam-se o Nó  $3_1$  e os parâmetros utilizados para o cálculo das distâncias. Nesta figura, adota-se o ponto B como referência. As distâncias foram calculadas em relação a este ponto.

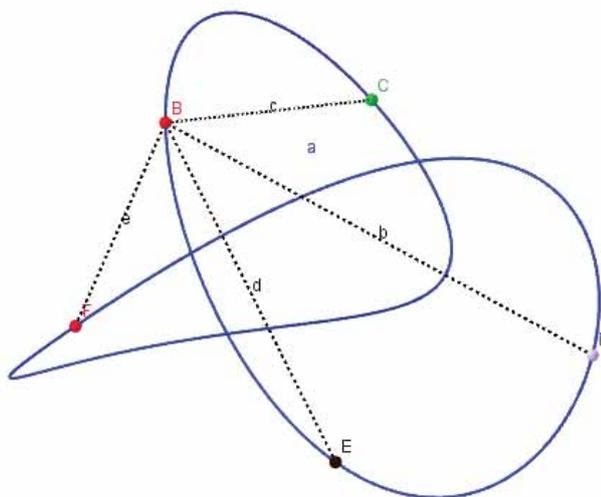


Figura 4 - Distâncias internas no Nó  $3_1$ . Fonte: Laboratório Simulação Molecular.

### 1.1 Proteínas

O termo proteína deriva do grego *proteios* que significa “primeiro” ou “a mais importante”. As proteínas são consideradas macromoléculas muito importantes das células e, para muitos organismos, constituem quase 50% de suas massas (CAMPBELL, 2001). Funcionam como catalisadores, transportam e armazenam outras moléculas, como o oxigênio, fornecem apoio mecânico e proteção imunitária, geram movimento, transmitem impulsos nervosos e

## FUNDAMENTAÇÃO TEÓRICA

---

controlam o crescimento e a diferenciação celular (BERG, STRYER E TYMOCZKO, 2008). As proteínas são hetero polímeros cujas unidades constituintes fundamentais são os aminoácidos naturais.

### 1.2 Aminoácidos

Os aminoácidos, por sua vez, são moléculas orgânicas que possuem ligados a um de seus átomos de carbono (denominado de carbono  $\alpha$ ) um átomo de hidrogênio, um grupo amina, um grupo carboxílico e uma cadeia lateral "R" específica para cada aminoácido. A cadeia lateral é o que difere os aminoácidos em sua estrutura, tamanho, cargas elétricas e solubilidade em água. Além de conferirem propriedades físico-químicas diferentes a cada aminoácido, as cadeias laterais são também responsáveis por forças estabilizadoras, advindas de interações fracas, que mantêm as estruturas conformacionais enoveladas das proteínas.

Os aminoácidos presentes nas moléculas de proteínas são ligados covalentemente uns aos outros por uma ligação denominada peptídica. Essa ligação é formada por uma reação de condensação entre o grupo carboxílico de um aminoácido e um grupo amina de outro aminoácido.

### 1.3 Estrutura Primária

As proteínas podem ser descritas em relação a diferentes conformações estruturais. A sequência de aminoácidos de uma proteína é designada como estrutura primária e é dada pelas ligações peptídicas do esqueleto covalente da molécula (CHAMPE, HARVEY E FERRIER, 2008). Consiste no nível estrutural mais simples e importante de uma proteína, pois lhe confere a sua identidade (ID) (MARZOCCO; TORRES, 2007).

### 1.4 Estrutura Secundária

O arranjo espacial entre aminoácidos adjacentes e próximos ao longo da cadeia é determinado pela estrutura secundária. Dois dos principais elementos de estruturas secundárias são hélice e folha, descritas abaixo:

- A hélice  $\alpha$  é o tipo mais comum de estrutura secundária. Caracteriza-se por uma estrutura helicoidal, estabilizada por ligações de hidrogênio estabelecidas entre os grupos C=O e NH de aminoácidos que distam quatro resíduos entre si.
- A folha  $\beta$  corresponde à estrutura estendida constituída por uma associação lateral de segmentos peptídicos em orientação paralela ou antiparalela, mantida por ligações de hidrogênio (HORTON, 1992).

### **1.5 Estrutura Terciária**

A estrutura terciária está relacionada ao enovelamento da cadeia proteica sobre ela mesma. Esta estrutura descreve o arranjo espacial global dos aminoácidos em uma proteína, podendo ocorrer contatos entre aminoácidos bastante separados ao longo da cadeia linear (HORTON, 1992; DAVID; COX, 2013).

### **1.6 Estrutura Quaternária**

Certas proteínas são constituídas por mais de uma cadeia proteica. A estrutura quaternária é o resultado da disposição espacial relativa de subunidades proteicas em proteínas complexas, formadas por duas ou mais cadeias polipeptídicas (CAMPBELL, 2001; VOET; VOET, 2006; MARZOCCO; TORRES, 2007; CHAMPE; HARVEY; FERRIER, 2008; BERG; STRYER; TYMOCZKO, 2008.).

### 1.7 Estudos sobre Nós (*Knots*) em Proteínas

Um dos grandes desafios da ciência é entender o mecanismo pelo qual uma proteína obtém e mantém a sua estrutura nativa tridimensional. (KRAFFT, 1931; HORTON, 1992; OUELLETTE E BAXEVANIS, 1998; HONIG, 1999; ANDRAE, 2006; VOET E VOET, 2006; QIAN *et al.*, 2007; KANG E KINI, 2009; MALLAM, 2009; DAVID E COX, 2013).

Após o descobrimento dos Nós (*knots*) em proteínas, o estudo de enovelamento proteico foi alçado a outros níveis de análise devidos, em grande parte, à necessária revisão de alguns modelos tidos como paradigmas na área, como o conceito de ordem de contato. De acordo com este conceito, representa-se o número médio de aminoácidos existentes ao longo da cadeia entre os pares de aminoácidos em contato topológico, normalizado pelo número total de contatos e pelo número de aminoácidos da cadeia. Contatos topológicos são definidos por pares de aminoácidos que se encontram no espaço de uma distância pequena (5-7 Å).

A maioria dos Nós presentes nas proteínas apresenta topologia  $3_1$  (trevo),  $4_1$  e  $5_2$ . Estas topologias são apresentadas na Figura 5 (FORGAN; SAUVAGE; STODDART, 2011; SULKOWSKA *et al.*, 2013).

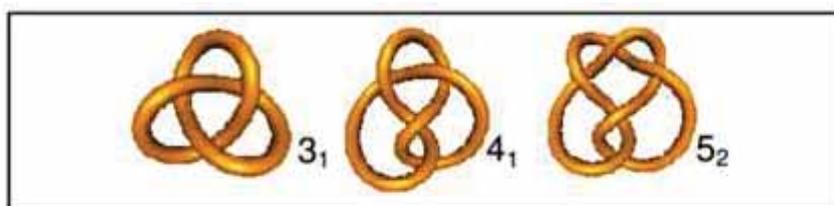


Figura 5 - Topologias associadas a Nós. Fonte: ADAMS, 2004.

## FUNDAMENTAÇÃO TEÓRICA

---

Conceitos de Nós apresentam importância significativa para a área biológica. Por exemplo, a doença de Parkinson está associada à Hidrolase de Ubiquitina (UCH-L1) ou o seu homólogo estrutural UCH-L3, uma proteína que contém os Nós  $5_2$  (*Knots*), considerados, segundo o atual conhecimento, como os mais complicados, e que estão relacionados à enzima na degradação da proteína (KONIARIS E MUTHUKUMAR, 1991; SACKHEIM E LEHMAN, 2001; MALLAM E JACKSON, 2005; MALLAM *et al.*, 2008; DZUBIELLA, 2009; DAY E THOMPSON, 2010; KING *et al.*, 2010; FORGAN, SAUVAGE E STODDART, 2011; GRACY E CHICHE, 2011; SULKOWSKA *et al.*, 2012; HSIEH *et al.*, 2013; RAWDON *et al.*, 2013; SULKOWSKA *et al.*, 2013).

O desafio inclui a compreensão do enovelamento de proteínas, mecanismo que ainda permanece um mistério (HONIG, 1999; MALLAM E JACKSON, 2005; KANG E KINI, 2009; VIRNAU, MALLAM E JACKSON, 2011; SKRBIĆ, MICHELETTI E FACCIOLI, 2012; SULKOWSKA *et al.*, 2012; SULKOWSKA *et al.*, 2013). A Figura 6 apresenta as topologias matemáticas existentes dos Nós, sendo a sua representação lida da seguinte forma:  $3_1$  possuem três cruzamentos e uma configuração, isto é, formas às quais podem se apresentar.

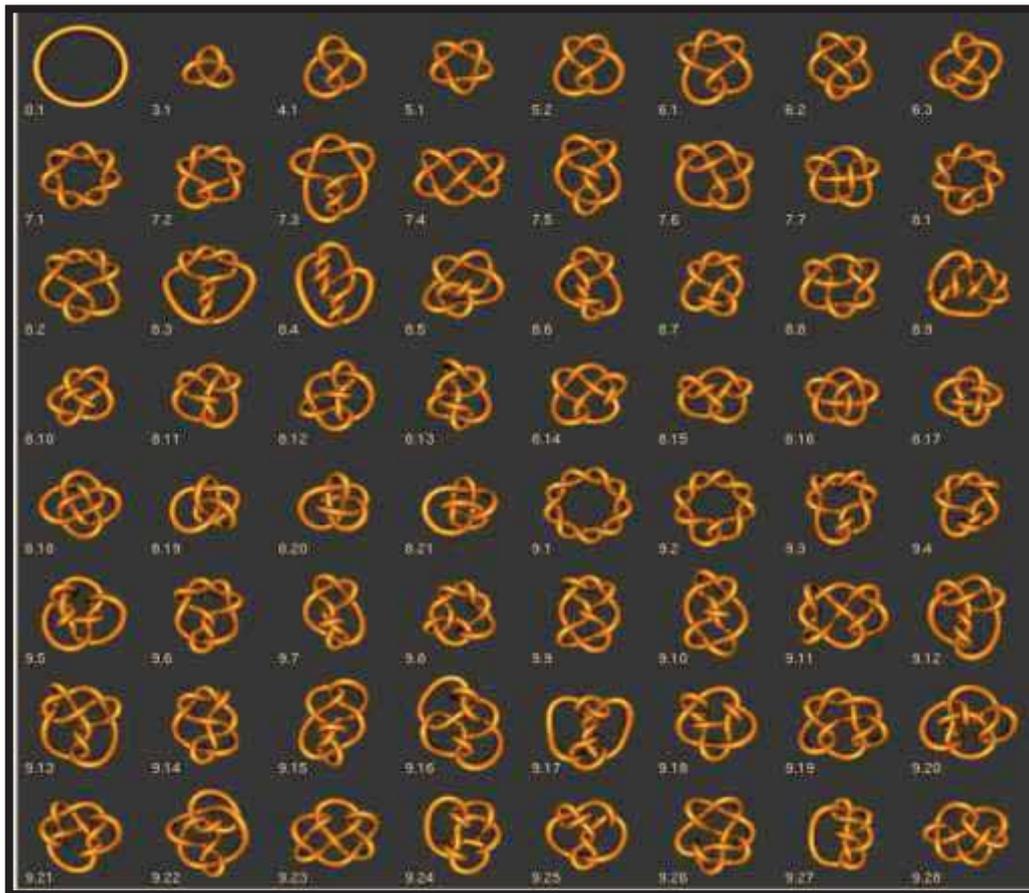


Figura 6 - Topologia dos Nós Matemáticos. Fonte: ADAMS, 2004.

### 1.8 Motivação e hipóteses de Pesquisa

A motivação originou-se no ano de 2007 ao realizar o estudo de um modelo simplificado de proteína em uma rede quadrada e cúbica, tratado por meio de simulações em Modelo Hidrofóbico Polar conhecido como HP, conforme Figura 7.

Ao estudar a expansão da rede cúbica para  $3 \times 3 \times 4$  encontrou-se, na simulação, a visualização do Nó em estruturas de proteínas (Figura 7c).

## FUNDAMENTAÇÃO TEÓRICA

Dessa forma, instigada pelo questionamento do que seriam estes Nós em proteínas, a pesquisa iniciou-se.

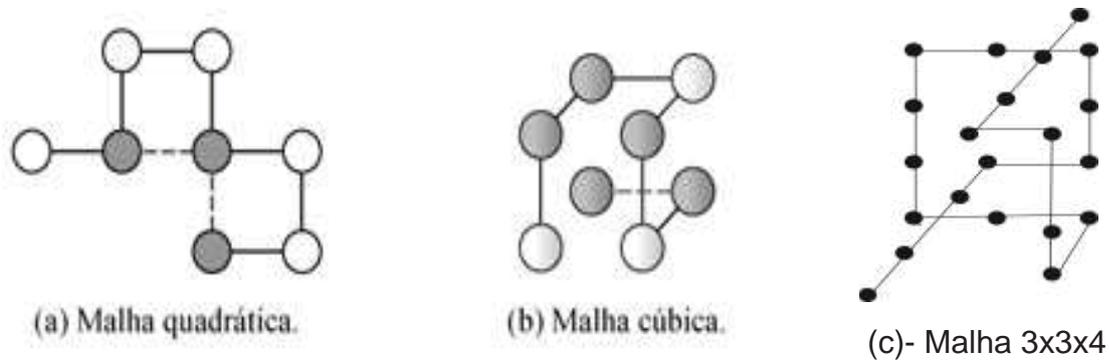


Figura 7 - Exemplo de conformações em malha quadrática (a) e malha cúbica (b) em Modelo HP. Esferas cinza ilustram os aminoácidos Hidrofóbicos e as brancas, os Polares. A malha 3x3x4 é apontada na (c). Fonte: The Self-Avoiding Walk, 2007.

### 1.9 Objetivos

O presente estudo tem como objetivo investigar padrões de comportamento em proteína real que possuam Nós matemáticos.

Como objetivos específicos, são apresentados os seguintes:

- Confirmar se as proteínas possuem Nós.
- Automatizar processo de cálculo de distância.
- Automatizar processo de cálculo de diedros.
- Verificar a hidrofobicidade nesta região.

## 2 TEORIA DOS NÓS

A Teoria dos Nós apresenta-se como uma área da Matemática que, inserida no campo da Topologia Algébrica, estuda as curvas no espaço, fechadas e sem autointerseções. De acordo com Adam (1994), pode-se dizer que as curvas são imagens de um círculo fechado no espaço tridimensional. O Nó da Matemática é uma curva no espaço, sem autointerseções e fechada em um círculo no espaço tridimensional.

Um Nó não é uma curva particular, mas todo o conjunto de posições que ela pode assumir se for deformada de acordo com determinados critérios, como projeções e movimentos Reidemeister (ELIAHOU; KAUFFMAN; THISTLETHWAITE, 2003).

O polinômio de Alexander foi um dos primeiros utilizados para cálculos de Nós. O polinômio de Jones é calculado sobre uma projeção do Nó, isto é, 3D para 2D, sendo a projeção da sua imagem achatada no plano, o que se verifica pelos diagramas dos Nós. Sendo assim, duas projeções quaisquer do mesmo Nó terão o mesmo polinômio de Jones, que se define como uma invariante isotópica (KAUFFMAN, 1990; ADAM, 1994; AUDOLY; CLAUVELIN; NEUKIRCH, 2007).

A invariante isotópica é utilizada para identificar projeções que são iguais. Se o resultado da invariante for diferente, então uma não poderá ser convertida na outra e, nesse caso, duas projeções com polinômios iguais não representarão o mesmo Nó. Contudo, diz-se que o polinômio de Jones não é

## FUNDAMENTAÇÃO TEÓRICA

---

uma invariante completa, pois ele não responde sobre a equivalência de duas projeções (KAUFFMAN, 1990).

Até os dias atuais não existe um polinômio invariante finito para o cálculo em um número finito de passos que seja completo (ADAM, 1994; DASBACH; LIN, 2006).

O Nó trivial, conforme Figura 8, é a classe de curvas que pode se deformar até se tornarem círculos (ELIAHOU; KAUFFMAN; THISTLETHWAITE, 2003).



Figura 8 - Representação Gráfica do Nó Trivial. Fonte: ADAMS, 2004.

### 2.1 Projeções

Projeções constituem a forma necessária de se representar um Nó por meio de um desenho no plano. É possível colocar o Nó sob um feixe de luz e observar a sombra que se projeta em um plano. Entretanto, desse modo,

## FUNDAMENTAÇÃO TEÓRICA

---

pode-se causar uma falsa ilusão do Nó, pois não se distingue que pedaços do Nó passam por cima ou por baixo de um cruzamento (ADAM, 1994; ELIAHOU; KAUFFMAN; THISTLETHWAITE, 2003; DASBACH; LIN, 2006).

Na Figura 9, observa-se que existe uma interrupção no desenho da curva pouco antes do cruzamento. Este Nó é conhecido como Trifólio. Nota-se que as projeções são regulares e cada cruzamento ocorre num só ponto. Para cada cruzamento verificam-se apenas dois segmentos de curva que se cruzam transversalmente e formam um X, resultando em um número de cruzamentos finito (ALEXANDER, 1928; ADAM, 1994; DASBACH E LIN, 2006; AUDOLY, CLAUVELIN E NEUKIRCH, 2007).

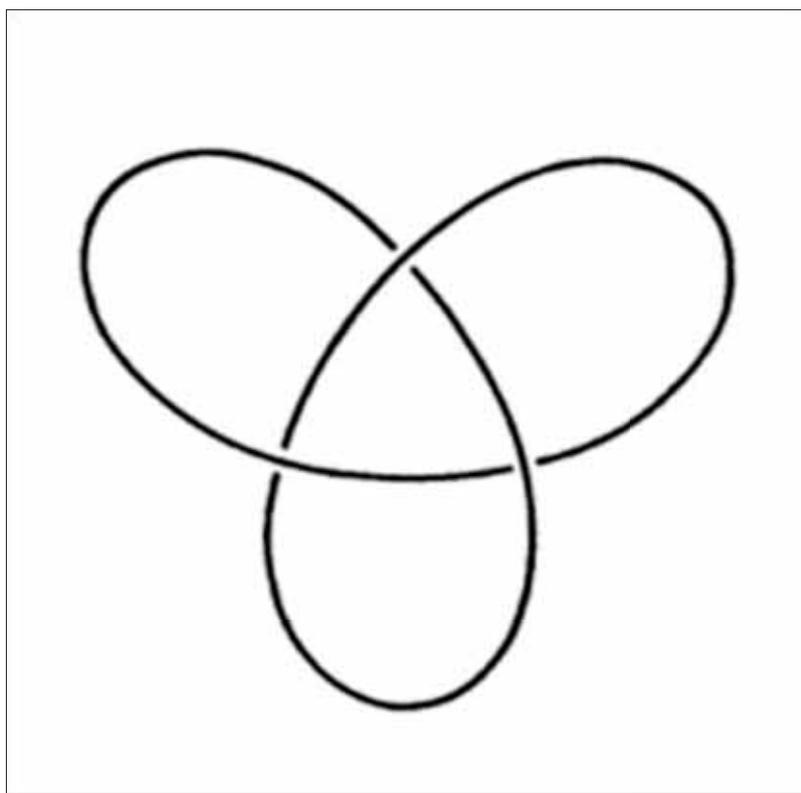


Figura 9 - Representação Gráfica do Trifólio. Fonte: ADAMS, 2004.

## FUNDAMENTAÇÃO TEÓRICA

---

Na Figura 9, verifica-se que a invariante isotópica pode desfazer cruzamentos ou mudar sua disposição, porém, não pode inverter a posição de dois segmentos em um cruzamento. Se assim fosse, qualquer Nó poderia alterar a posição do segmento possibilitando à curva tornar-se um círculo.

Deste modo, consideram-se duas projeções equivalentes se uma invariante isotópica transforma-se em outra (ALEXANDER, 1928; ADAM, 1994; KATRITCH *et al.*, 1997; MELUZZI, SMITH E ARYA, 2010; SULKOWSKA *et al.*, 2013).

Um Nó é representado, portanto, por uma classe de projeções (regulares) possíveis, todas equivalentes entre si (ADAMS, 1994; MALLAM, 2009).

### 2.2 Movimentos de Reidemeister

Há vários tipos de invariantes e nem todas são calculadas sobre projeções. Kurt Reidemeister afirmou que toda isotopia entre duas projeções regulares pode ser efetuada com apenas três tipos de transições (ADAM, 1994). A estas transições denominou de Movimentos de Reidemeister. Portanto, precisam-se testar apenas três tipos de movimentos para saber se um polinômio se altera ou não entre as isotopias (ALEXANDER, 1928).

Movimentos tipo I são mostrados na Figura 10, onde se observa um laço simples com a presença de duas variantes, sendo uma imagem especular da outra (ADAMS, 1994).

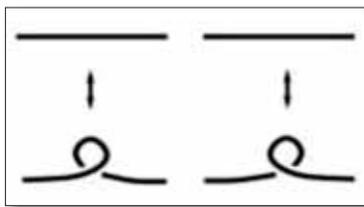


Figura 10 - Movimento Reidemeister 01. Fonte: ADAMS, 2004.

Movimentos tipo II necessitam de dois segmentos e possuem duas variantes. Na transição, existem dois segmentos que se tangenciam, como mostra a Figura 11 (ADAMS, 1994).

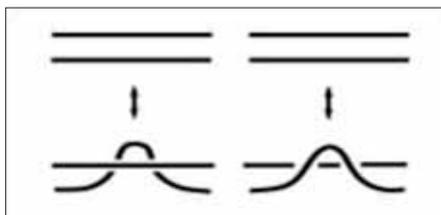


Figura 11 - Movimento Reidemeister 02. Fonte: ADAMS, 2004.

Movimentos tipo III envolvem três segmentos. A transição é um cruzamento onde estes segmentos concorrem (Figura 12). Considera-se como horizontal o segmento que esteja por cima dos outros dois (ADAMS, 1994).

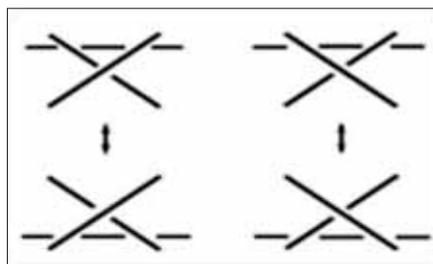


Figura 12 - Movimento Reidemeister 03. Fonte: ADAMS, 2004.

Uma das questões em teoria dos Nós é saber quando dois deles representam um mesmo Nó. Para saber se dois Nós são diferentes, calcula-se seus invariantes, isto é, expressões algébricas ou numéricas a eles associadas que não variam quando o Nó é deformado.

### 2.3 Invariante Determinante

Para definir um Nó, é necessário calcular o Determinante D. Para isso, associa-se a cada arco da Figura 13 uma variável e a cada cruzamento, uma equação da forma.

$$x+y-2z = 0$$

Equação 1

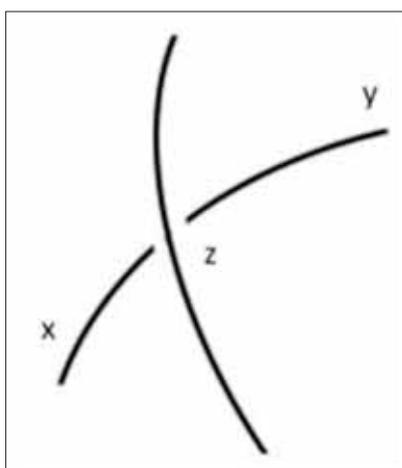


Figura 13 - Representação gráfica dos cruzamentos. Fonte: ADAMS, 2004.

## FUNDAMENTAÇÃO TEÓRICA

A seguir, coloca-se uma variável qualquer igual a zero e descarta-se uma equação também qualquer, isto é, determina-se um sistema de  $n-1$  equações e  $n-1$  variáveis, onde  $n$  é o número de cruzamentos. Calcula-se, então, o valor absoluto do Determinante da Matriz formada pelos coeficientes das equações do sistema linear, como se observa na Figura 14.

Equações:

$$\mathbf{x+z-2y=0; \quad \text{Equação 2}}$$

$$\mathbf{x+y-2z=0; \quad \text{Equação 3}}$$

$$\mathbf{z+y-2x=0; \quad \text{Equação 4}}$$

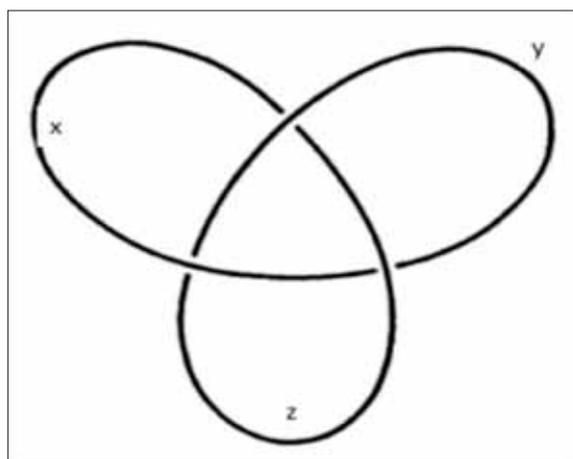


Figura 14 - Equações do sistema linear. Fonte: ADAMS, 2004.

Coloca-se a variável  $x=0$  e, eliminando a primeira equação, temos o sistema:

$$\begin{cases} y - 2z = 0 \\ y + z = 0 \end{cases} \quad \text{Equação 5}$$

Em seguida, calcula-se o Determinante:

$$\begin{vmatrix} 1 & -2 \\ 1 & 1 \end{vmatrix} = 3 \quad \text{Equação 6}$$

$$\text{Assim, temos que } D = |3| = 3 \quad \text{Equação 7}$$

## 2.4 Invariante Polinômio Alexander

Baseados na literatura ao longo do século, foram criados vários processos que permitiram classificar alguns Nós. O Polinômio de Alexander é um invariante de Nós descoberto, em 1923, por James Waddell Alexander (KAUFFMAN, 1990; ADAM, 1994).

Neste processo, associa-se a cada Nó um polinômio, de modo que, se um Nó puder se transformar em outro, os seus respectivos polinômios serão iguais. Entretanto, deve-se ressaltar que tal polinômio não permite realizar a total distinção entre os Nós, uma vez que dois destes podem estar associados ao mesmo polinômio e não serem ambientes isotópicos (ALEXANDER, 1928; ADAM, 1994; KATRITCH *ET AL.*, 1997; ANDRAE, 2006; BRASHER, SCHAREIN E VAZQUEZ, 2013; RAWDON *et al.*, 2013).

Durante anos, o Polinômio de Alexander foi o único a descrever os enlaces orientados. Este polinômio está baseado no cálculo do determinante de uma matriz associada a um diagrama de um enlace (ADAM, 1994). Para a sua determinação, é necessário identificar os vários cruzamentos e arcos do diagrama K que o representa (ADAM, 1994). Os cruzamentos são identificados por números naturais e o arcos por letras minúsculas. Assim, define-se uma

## FUNDAMENTAÇÃO TEÓRICA

matriz  $n \times n$ , onde  $n$  é o número de cruzamentos do diagrama  $K$  (ALEXANDER, 1928; ADAM, 1994). A cada linha da matriz associa-se um cruzamento e a cada coluna, um arco, definindo-se a matriz de acordo com o processo representado na Figura 15.

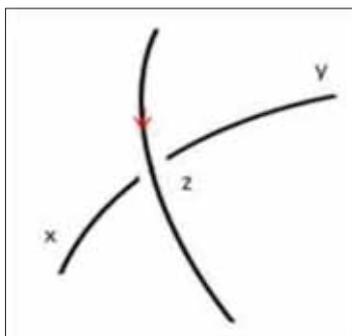


Figura 15 - Orientação do diagrama K. Fonte: ADAMS, 2004.

Para cada cruzamento escreve-se uma equação, conforme se observa na Figura 15, em que  $z$  deve ser identificado como a variável que passa superiormente na região do cruzamento. Na Figura 15, para a escolha de  $x$  e  $y$  na equação, usa-se a orientação do trecho superior do Nó no cruzamento. Assim, a variável  $x$  identifica-se com a variável associada à direita de  $z$  e o  $y$ , com a variável à esquerda de  $z$  (ADAMS, 1994).

Seja um polinômio  $\Delta(t)$  da forma:  $x_n t^n + x_{n-1} t^{n-1} + \dots + x_0 + \dots + x_{-m} t^{-m}$ , onde os coeficientes  $x_i$  são inteiros e a variável  $t$  satisfaz  $t^p t^q = t^{p+q}$

A equação 8 é representada abaixo:

$$y - tx - (1-t)z = 0; \quad \text{Equação 8}$$

Para encontrar as equações do Nó trifólio definimos a sua orientação conforme a Figura 16.

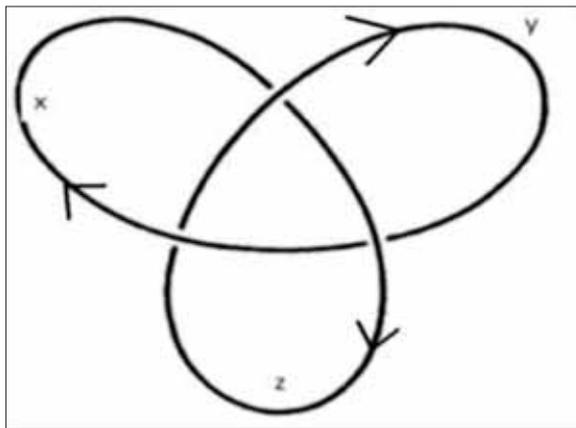


Figura 16 - Orientação definida. Fonte: ADAMS, 2004.

As equações definidas de acordo com a orientação da Figura 16 são:

$$x - tz - (1 - t)y = 0 \quad \text{Equação 9}$$

$$y - tx - (1 - t)z = 0 \quad \text{Equação 10}$$

$$z - ty - (1 - t)x = 0 \quad \text{Equação 11}$$

Inicia-se com  $x=0$  e elimina-se a primeira equação, obtendo-se o sistema descrito abaixo:

$$\begin{cases} y - (1 - t)z = 0 \\ -ty + z = 0 \end{cases} \quad \text{Equação 12}$$

Assim, tem-se que:

$$\delta(t) = \begin{vmatrix} 1 & -1 + t \\ -t & 1 \end{vmatrix} = 1 - t + t^2 \quad \text{Equação 13}$$

Por fim, para que  $\Delta(t) = \Delta(t^{-1})$  e  $\Delta(1) = +1$ , multiplica-se  $\delta(t)$  por  $t^{-1}$ , obtendo-se:

$$t^{-1}\delta(t) = t^{-1} + t = \Delta(t) \quad \text{Equação 14}$$

## 2.5 Distância Interna

O termo distância está presente em todos os conceitos da Geometria Analítica, pois, nesta área da matemática, tem-se a relação entre elementos geométricos e algébricos e como elemento básico o ponto (LUA E GROSBERG, 2006; JIANG *et al.*, 2013; SOUZA *et al.*, 2013). Este recebe coordenadas no plano cartesiano e, por meio destas, pode-se encontrar o valor da distância entre dois pontos no espaço.

As coordenadas internas, apresentadas nos arquivos PDB, permitem a determinação direta de diversos parâmetros estruturais, tais como distâncias, ângulos e ângulos de torção. Para o cálculo da distância entre dois carbonos  $\alpha$ ,  $i$  e  $j$ , de coordenadas X, Y e Z, usa-se a seguinte equação:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

Equação 15

Onde:

D – distância interna

	X <sub>1</sub>	Y <sub>1</sub>	Z <sub>1</sub>								
ATOM	2	CA	PHE	A	6	-1,586	32,024	2,933	1,00	36,15	C
ATOM	13	CA	MET	A	7	-3,835	33,278	-0,383	1,00	34,92	C

Figura 17 - Coordenadas cartesianas para o cálculo de distância.  
Fonte: Laboratório Simulação Molecular.

## FUNDAMENTAÇÃO TEÓRICA

A relação utilizada para calcular as distâncias internas dos aminoácidos carbono  $\alpha$  no espaço tridimensional é dada pela Equação 15.

Calcula-se o valor D na Equação 15 substituindo as coordenadas X, Y e Z dos aminoácidos carbono  $\alpha$  de dois pontos diferentes, ou seja, ATOM 13 e ATOM 2, vistos na Figura 17.

Usando-se as coordenadas cartesianas de um arquivo PDB para um par de átomos, tem-se a distância em Å. Podem-se analisar os carbonos  $\alpha$  da proteína 2efv na Figura 18, onde se representa como é feito o rastreo do cálculo de distância para a primeira posição do carbono  $\alpha$  que se inicia Ca2.

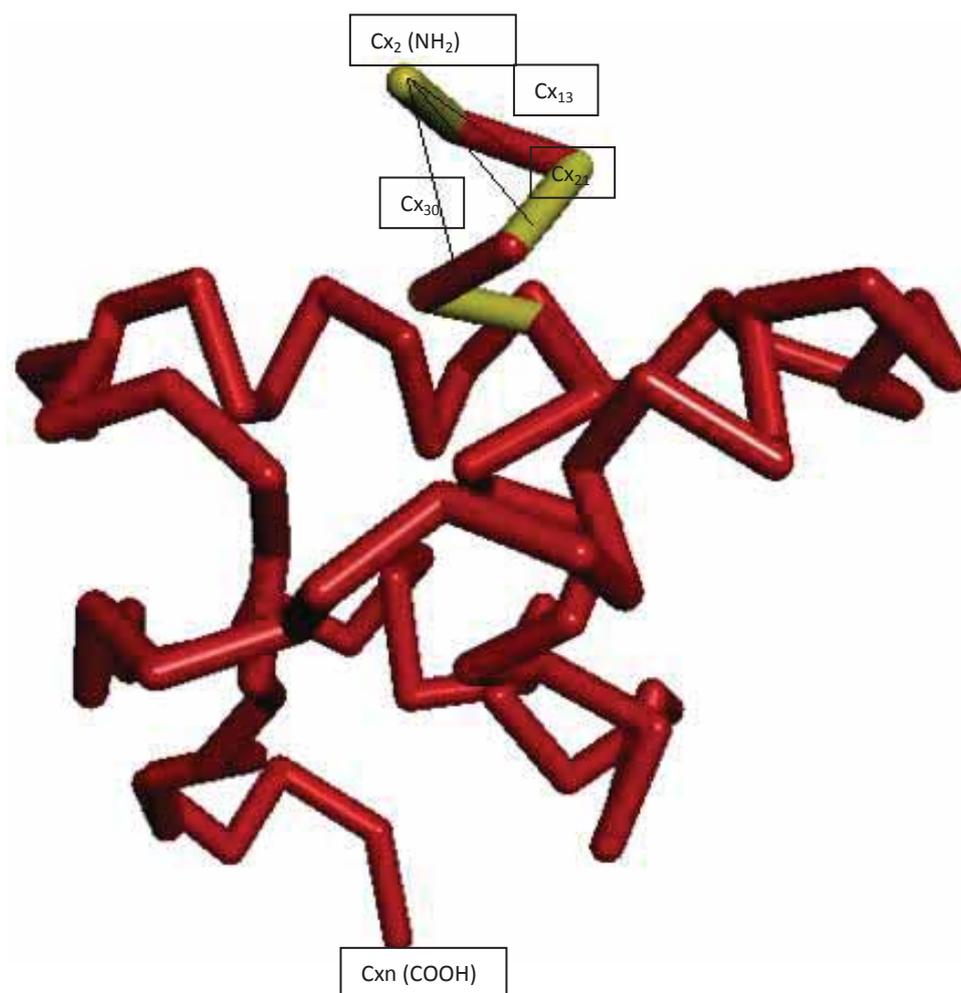


Figura 18 - Carbonos alfa da proteína 2efv e suas distâncias internas em ångström (Å).  
Fonte: Laboratório Simulação Molecular.

## 2.6 Diedros

Um ângulo diedro de uma proteína é o ângulo interno do esqueleto polipeptídico onde dois planos adjacentes se encontram. A formação dos planos, na Figura 19, ilustra o primeiro plano formado pelos carbonos  $\alpha$  1, 2 e 3. O segundo plano, por sua vez, é formado pelos carbonos  $\alpha$  2, 3, 4. Após a formação dos planos, determina-se a Equação 16 para cada um deles.

$$A_1x + B_1y + C_1z + D_1 = 0 \quad \text{plano 1 } C\alpha_1, C\alpha_2, C\alpha_3 \quad \text{Equação 16}$$

$$A_2x + B_2y + C_2z + D_2 = 0 \quad \text{plano 2 } C\alpha_2, C\alpha_3, C\alpha_4$$

Conhecendo-se a equação dos dois planos, calcula-se o ângulo diedral. Observe a Equação 17:

$$\cos \alpha = \frac{A_1A_2 + B_1B_2 + C_1C_2}{\sqrt{A_1^2 + B_1^2 + C_1^2} \sqrt{A_2^2 + B_2^2 + C_2^2}} \quad \text{Equação 17}$$

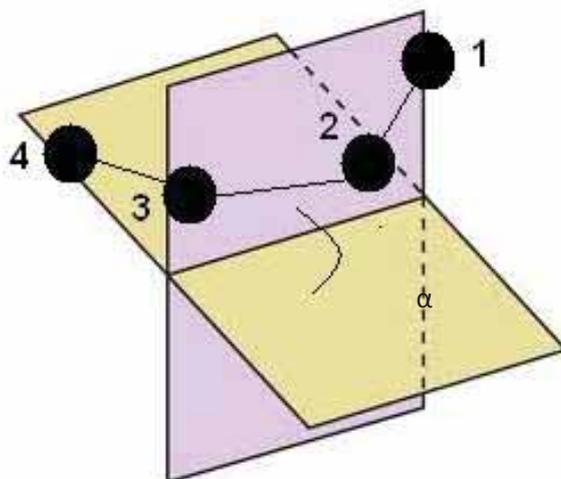


Figura 19 - Ângulo entre dois planos. Fonte: Laboratório Simulação Molecular.

## 2.7 Distância Ponta – Ponta

A Distância Ponta-Ponta de uma proteína é definida como a distância entre os dois aminoácidos  $C\alpha$  das extremidades da cadeia, a qual pode variar de uma proteína para outra (MOGLICH, JODER E KIEFHABER, 2006; ZHANG *et al.*, 2012).

Utilizando-se de vetor para representar a proteína, conforme apresentado na Figura 20, observa-se em pontilhado o vetor  $\vec{R}$  que especifica a distância entre o início e o fim da cadeia.

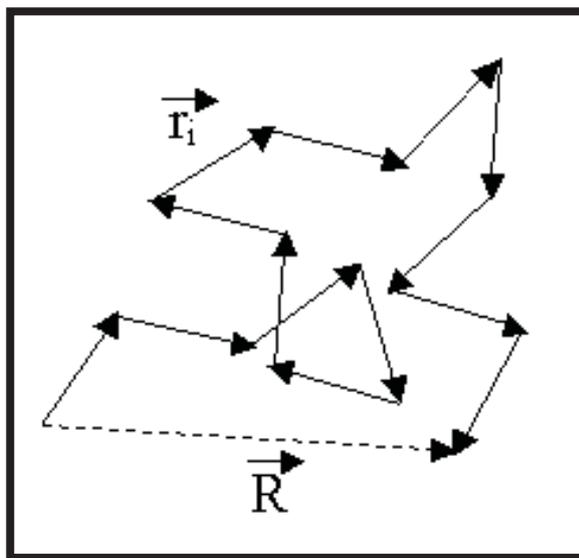


Figura 20 - Distância final-final. Fonte: Laboratório Simulação Molecular.

## 2.8 Banco de Dados

O *Protein Data Bank* (PDB) foi criado em 1971 como um site para manter arquivos de estruturas cristalinas de macromoléculas biológicas. Na década de 1990, o número de estruturas depositadas começou a aumentar drasticamente em decorrência do aperfeiçoamento da tecnologia em todos os aspectos do processo de cristalografia e da adição de estruturas determinadas por métodos de ressonância magnética nuclear (RMN) (KOLESOV *et al.*, 2007).

A utilização inicial do PDB era limitada a um pequeno grupo de especialistas envolvidos na investigação estrutural. Hoje, depositantes para o PDB têm diferentes conhecimentos nas técnicas de raios-X, RMN e microscopia eletrônica (PDB, 2013). Observe a Figura 21:

The image shows the PDB website interface. At the top, it says 'PDB PROTEIN DATA BANK' and 'An Information Portal to Biological Macromolecular Structures'. Below this, there's a search bar and navigation links. The main content area is titled 'A Resource for Studying Biological Macromolecules'. It features a 'Molecule of the Month' section for Tuberculosis and a 'PDB Featured Molecule' section for Sugar coating the surface: yeast Alg13. The right sidebar contains 'News', 'Complete News', 'Newsletter', 'Discussion Forum', and 'Job Listings'. The bottom of the page has a 'New user?' section with links for browser compatibility and getting started.

Figura 21 - Banco de Dados. Fonte: PDB, 2013.

## FUNDAMENTAÇÃO TEÓRICA

---

Bancos de dados desempenham papel essencial nos estudos das estruturas de proteínas, pois constituem a fonte básica de informações para a predição, a análise e o estudo das estruturas.

Essa base de dados tem crescido consideravelmente, o que se pode notar a partir da análise dos dados relativos aos últimos 22 anos, conforme apresentando na Figura 22. Em 1991, o PDB continha apenas 695 estruturas; em 2001, já eram 16.430 estruturas, o que denota um crescimento de 2.364% em relação a 1991. Em 2013, segundo dados acessados no mês de novembro do corrente ano, já são mais de 90.000 estruturas. Esse crescimento exponencial do volume total de dados está representado pela cor vermelha na Figura 22.

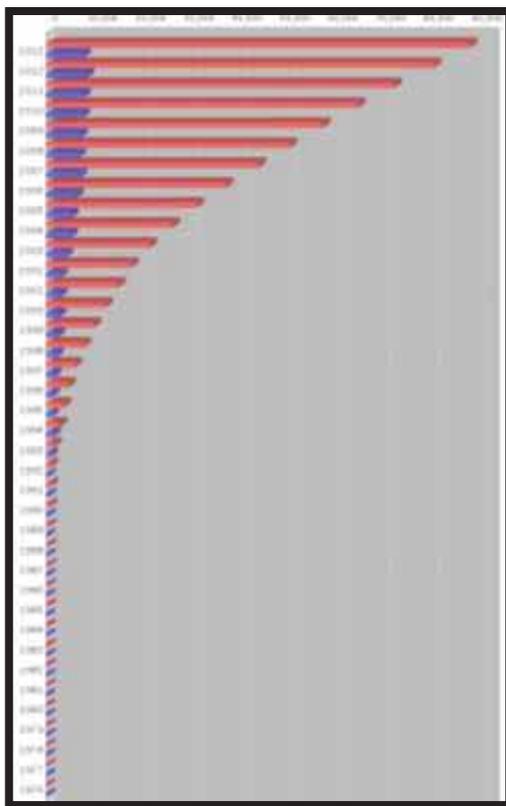


Figura 22 - Crescimento anual do número total de estruturas 3D de proteínas no PDB. Dados acessados em novembro de 2013. O azul ilustra a quantidade de estruturas determinadas anualmente e o vermelho, o total. Fonte: PDB, 2013.

### 3 MATERIAL E MÉTODOS

Para melhor compreender o sistema desenvolvido e ter uma visão geral do problema, é pertinente apresentar uma descrição dos materiais, dos procedimentos e da metodologia empregados para a obtenção dos resultados.

O material utilizado envolveu 40 estruturas das proteínas que possuem Nós cadastradas no PDB. Porém, para apresentação dos resultados nesta tese, foram selecionadas seis estruturas cujos códigos são: 1ns5, 1ztu, 2efv, 2etl, 3bjx e 2lyz esta última não possui Nó.

No laboratório de simulação computacional da Universidade Estadual Paulista (UNESP), campus de Bauru - SP, foi utilizado um microcomputador Pentium com 1GB de RAM. Os softwares empregados foram:

- Oringin 7.0 (Originlab, 2013).
- Pacote da Microsoft Office (Microsoft, 2013).
- CodeBlocks que é um ambiente de desenvolvimento integrado (IDE) para plataformas Linux, MAC OS e Windows, que permite escrever aplicativos para ambiente gráficos ou aplicativos de console em linguagem C ou C++, com suporte a múltiplos compiladores como GCC/MinGW, SDCC, Intel C++, GNU ARM, entre outros (Kernighan e Ritchie, 1988; Code::Blocks, 2013; Kohl, 2013; Summit, 2013).
- RasMol para visualização das proteínas (RasMol and OpenRasMol, 2013).
- KnotPlot para investigação dos Nós matemáticos (The KnotPlot Site, 2013).

A seguir, é apresentada a descrição dos métodos dividida em três etapas: levantamento, tratamento e integração dos dados.

### 3.1 Levantamentos de Dados

Após uma pesquisa bibliográfica, o procedimento para a obtenção das proteínas consiste de uma varredura exaustiva no Banco de Dados em busca de proteínas com uma característica peculiar que são os Nós em sua conformação tridimensional.

Esta varredura foi feita manualmente. A partir dela, levantaram-se todas as proteínas com Nós encontradas até o momento, perfazendo um total de 40 proteínas. Contudo, foram selecionadas apenas seis classificações distintas para detalhamento na tese (KOLESOV *ET AL.*, 2007; VIRNAU, MALLAM E JACKSON, 2011; BRASHER, SCHAREIN E VAZQUEZ, 2013; RAWDON *et al.*, 2013; SZYMCZAK, 2013; VIRNAU, 2013).

Depois de realizar o *download* dos arquivos do PDB de cada proteína selecionada, confirmaram-se os Nós pelo servidor de *web* disponível gratuitamente: <http://knots.mit.edu/> (VIRNAU, 2013). Verificou-se, também, a topologia dos Nós, o tamanho e o intervalo de sua formação.

O servidor *web* gratuito é um site disponível utilizado para detectar e confirmar os Nós em estrutura de proteínas. A implementação atual do servidor possui a modelagem matemática baseada no Polinômio de Alexander. Conforme demonstrado na Figura 23, o programa permite gerar uma visualização gráfica estática, além de uma interatividade em 3D.

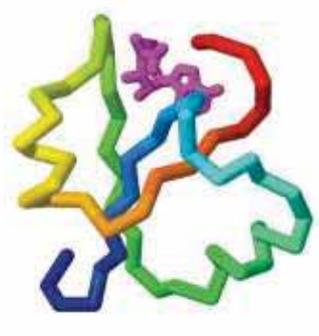
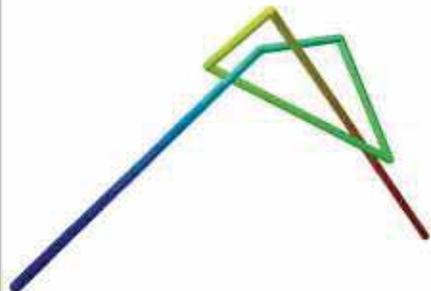
O Banco de dados do servidor oferece uma lista de Nós encontrados nas diversas classificações de proteínas que estão em constante atualização.

**A Knots found in the [1uam](#) structure:**

Knot residues	Chain start-stop	Knot type	Knot	
86-130A	-1-250A	<sup>31</sup> (trefoil knot)		<a href="#">Jmol visualization</a>

[Download results and rasmol scripts as zip package](#)

**B Residues 86-130A**

Knot in the <a href="#">1uam</a> structure	Simplified representation of the knot
	

[Hide/Show unknotted structure](#) [Spin the structures](#)

Hint: hold Ctrl-Alt to move the structure. Shift to zoom. Right click to get console.  
Knotted region is defined as 'knot', typing 'select knot' will select corresponding residues.

Enter one-line RasMol/Chime script commands here:

Figura 23 - Tela final do servidor. Nós por H. influenzae (PDB ID 1uam).<sup>1</sup>

<sup>1</sup> (A) Tabela de resumo. (B) Jmol visualização interativa. A estrutura 1uam é exibida na janela à esquerda com um Nó destacado em cores do arco-íris e restante da proteína. Neste caso, o Nó de trevo abrange uma região relativamente pequena da proteína e pode ser facilmente visto em sua estrutura. Em muitos casos, porém, a visualização é difícil. O painel direito apresenta a representação simplificada (reduzida) do Nó. Essas visualizações também podem ser ocultadas usando *scripts* RasMol previstos no pacote para *download*.

### 3.2 Tratamento dos Dados

Após a checagem das informações anteriores, utilizou-se o arquivo da estrutura da proteína para gerar uma listagem dos C $\alpha$  (carbono alfa). Para tanto, empregou-se o *software* CodeBlocks, um ambiente de desenvolvimento integrado, criado no Laboratório de Simulação Molecular da UNESP de Bauru. Para o desenvolvimento desse *software*, escolheu-se a linguagem C.

Realizou-se a implementação do *software* para a automação dos cálculos das distâncias internas e dos diedros, bem como o arquivo de rastreamento dos C $\alpha$  de cada estrutura.

A apresentação do *software*, mostrada na Figura 24, está no modo usuário para gerar os resultados que serão discutidos no próximo capítulo.

O intuito do *software* é rastrear exhaustivamente o arquivo depositado no PDB para selecionar todos os carbonos alfa da proteína, com o objetivo de obter arquivos externos com extensão .txt's com os respectivos resultados de otimização dos cálculos das distâncias internas e dos diedros.

A simulação será realizada com a estrutura 1nbs. O *software* é executado no ambiente que emula o prompt do comando do DOS pelo nome aminoacidos.exe. Após a execução do comando, apresenta-se a Figura 24, esperando que seja digitada a identificação da estrutura.

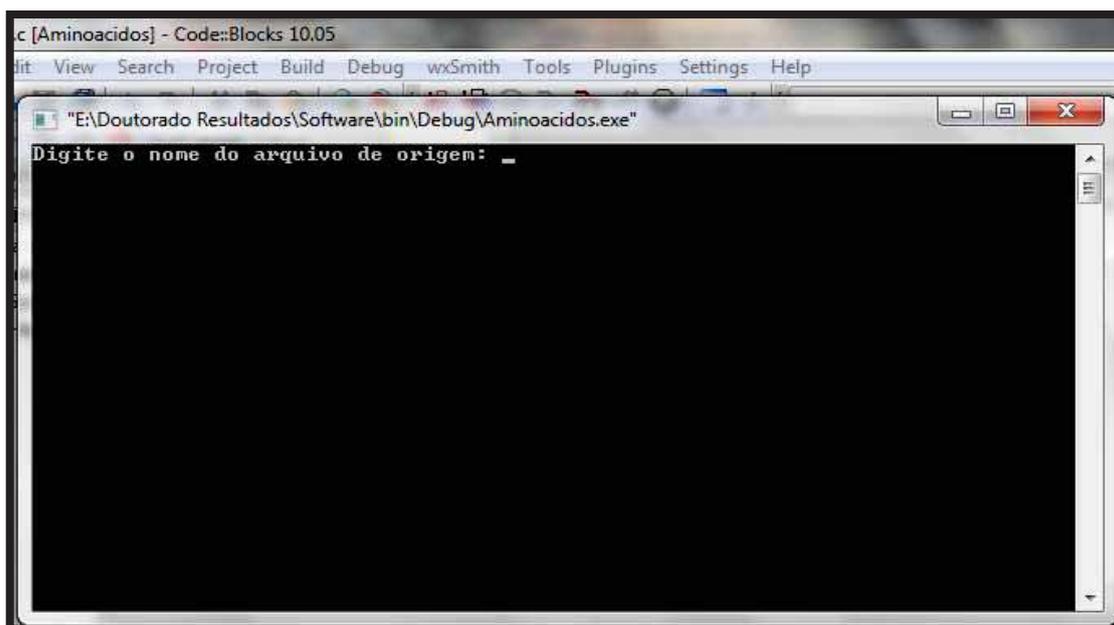


Figura 24 - Tela inicial do software executado. Fonte: Laboratório Simulação Molecular.

A Figura 25 identifica a estrutura 1nbs que deverá estar em arquivo na pasta onde o sistema aminoácidos.exe foi executado.

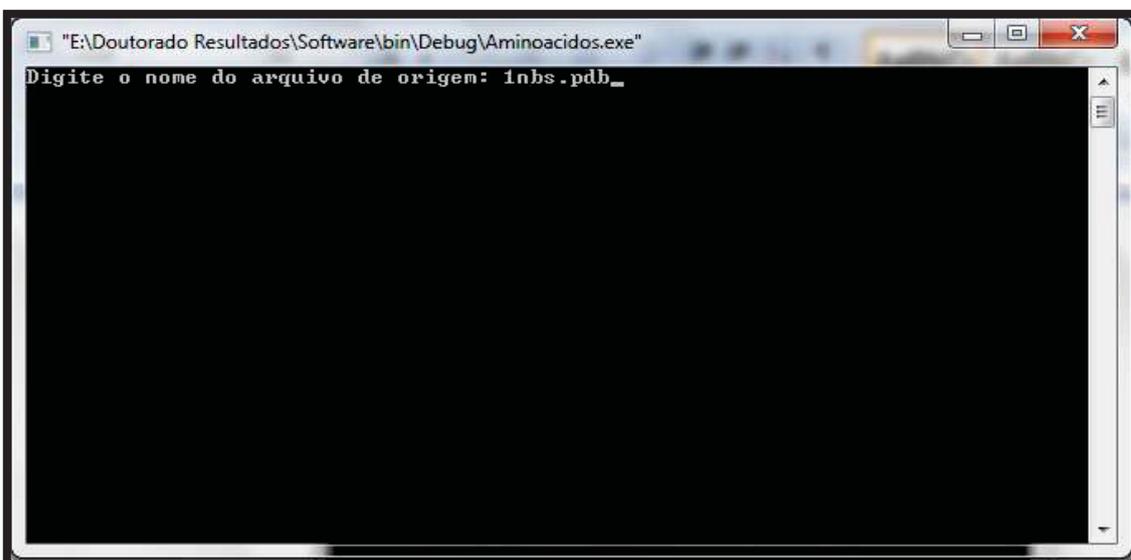


Figura 25 - Tela inicial identificando a estrutura a ser executada. Fonte: Laboratório Simulação Molecular.

Assim, os arquivos são gerados com sucesso como mostra a Figura 26.

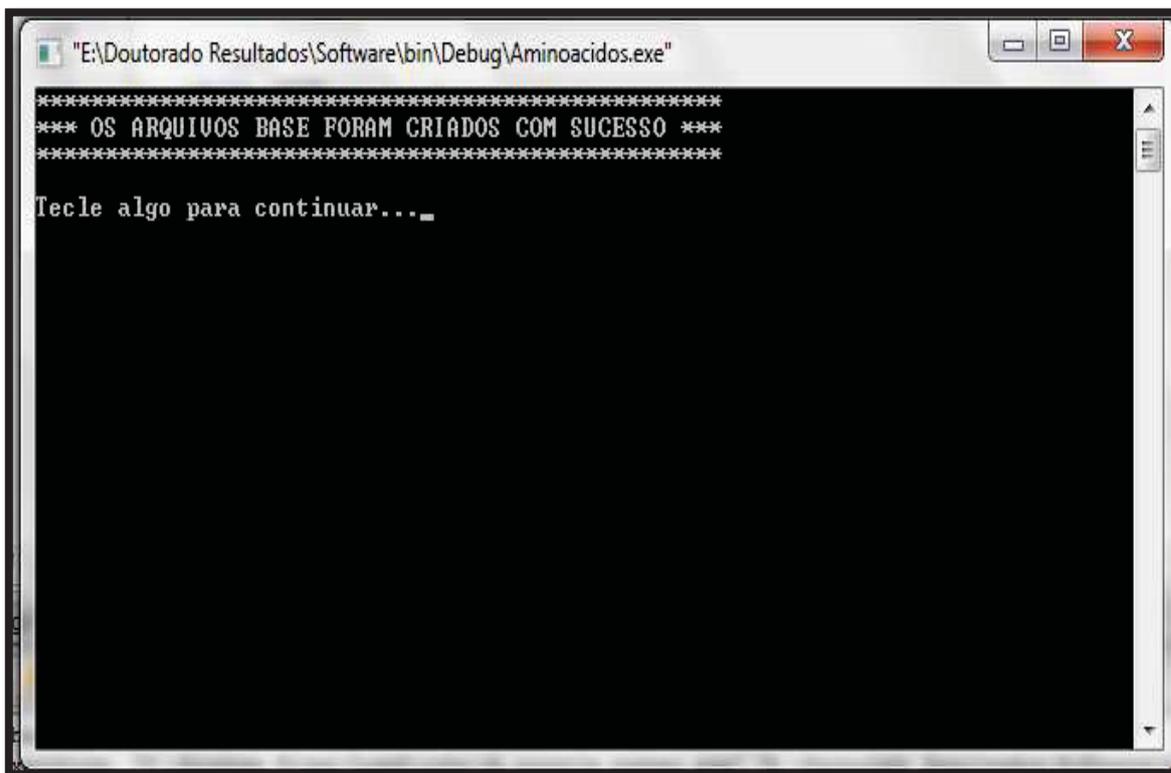


Figura 26 - Resultado executado corretamente. Fonte: Laboratório Simulação Molecular.

A Figura 27 identifica os arquivos gerados na mesma pasta. Apresenta-se um arquivo rastreado com todos aqueles que estão depurados da seguinte forma:

- base.txt - arquivo gerado da linha inteira de todos os átomos de carbono.
- coordenadas.txt - somente as coordenadas de todos os átomos de carbono.
- atomos\_CA.txt - identificação dos átomos de carbono alfa e suas coordenadas.



atomos_CA	25/06/2013 17:42	Arquivo TXT	25 KB
base	25/06/2013 17:42	Arquivo TXT	14 KB
coordenadas	25/06/2013 17:42	Arquivo TXT	14 KB

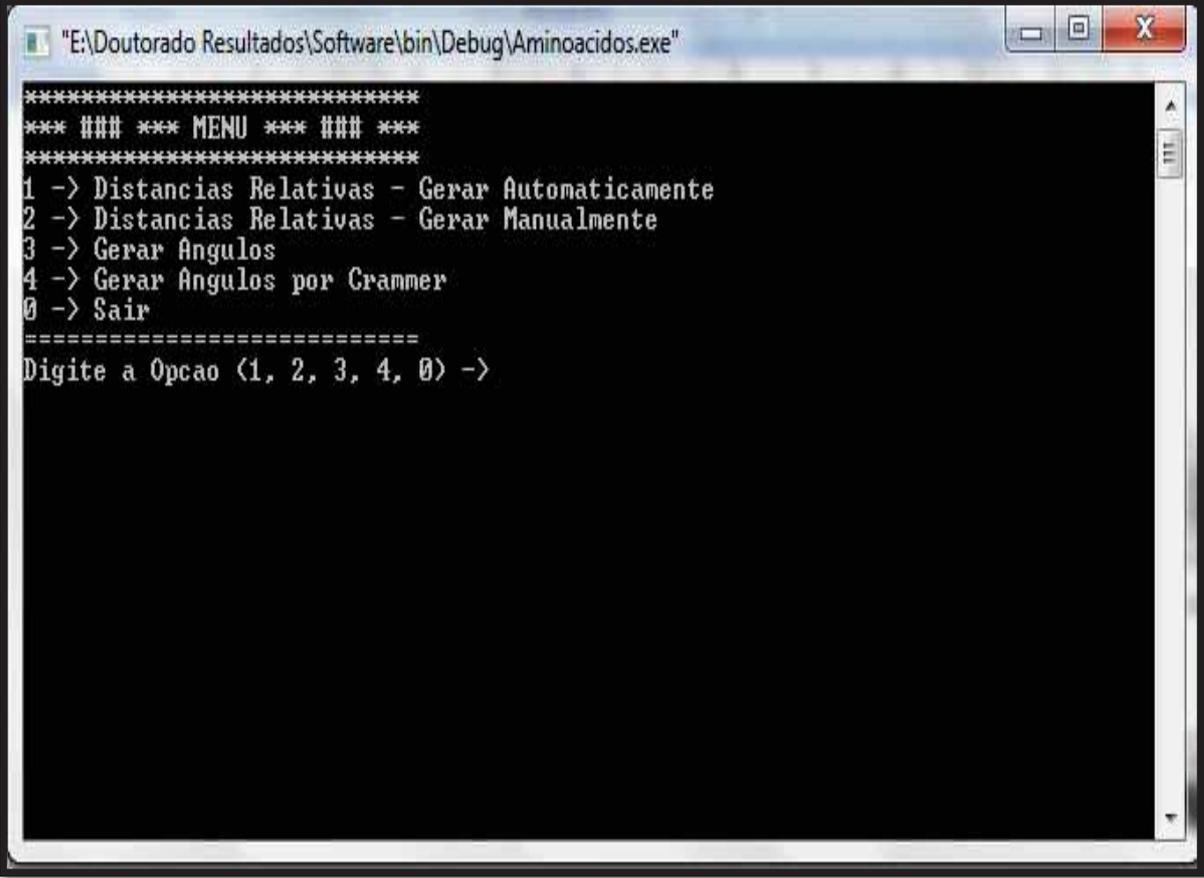
Figura 27 - Arquivos que serão utilizados para cálculos. Fonte: Laboratório Simulação Molecular.

Após gerar os arquivos acima, utiliza-se o mesmo para os cálculos.

A Figura 28 representa o Menu que é acionado para a escolha das opções:

- **Opção 01:** realiza os cálculos das distâncias automaticamente, isto é, se o arquivo com extensão .txt foi gerado com sucesso para a proteína 1nbs.pdb, teremos 153 átomos de carbono. Portanto, a obtenção dos resultados deste cálculo serão 153 arquivos com extensão .txt gerados automaticamente com os resultados das distâncias internas de cada uma das proteínas. O *software* foi desenvolvido para rastrear as distâncias internas de todas as posições dos carbonos alfa.
- **Opção 02:** realiza os cálculos das distâncias manualmente, especificando a posição em que se encontra o átomo de carbono alfa. Nesta opção, podemos gerar os cálculos das distâncias internas somente para a posição que desejarmos.
- **Opção 03:** calcula os ângulos entre os carbonos alfa.
- **Opção 04:** calcula os diedros entre os planos dos carbonos alfa pela Regra de Crammer.

- **Opção 0:** sair – encerra as atividades do *software*.



```
"E:\Doutorado Resultados\Software\bin\Debug\Aminoacidos.exe"
*****
*** ### *** MENU *** ### ***
*****
1 -> Distancias Relativas - Gerar Automaticamente
2 -> Distancias Relativas - Gerar Manualmente
3 -> Gerar Angulos
4 -> Gerar Angulos por Cramer
0 -> Sair
=====
Digite a Opcao (1, 2, 3, 4, 0) ->
```

Figura 28 - Menu Inicial. Fonte: Laboratório Simulação Molecular.

A Figura 29 detalha a tela após a escolha de qualquer uma das opções acima.

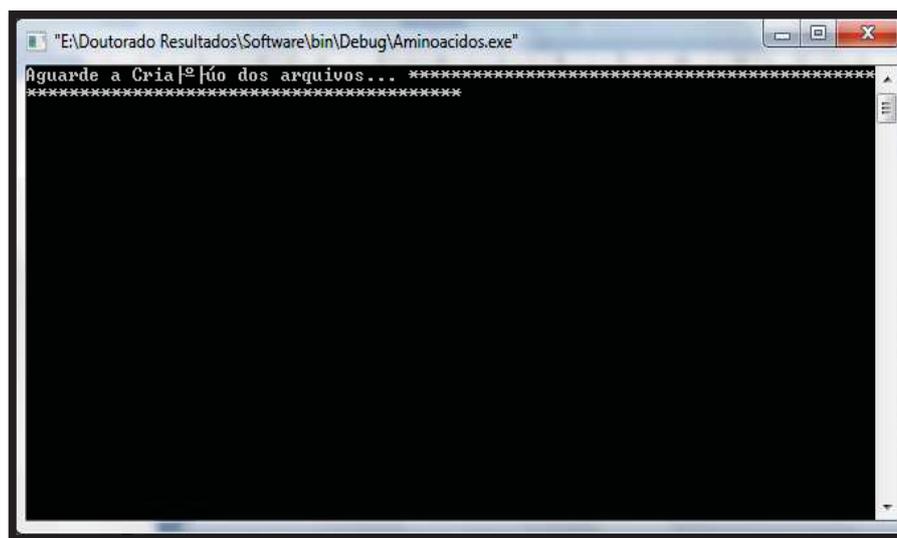


Figura 29 - Criação dos arquivos de qualquer opção

Ao final de qualquer uma das opções anteriores, para o usuário saber se os arquivos foram gerados corretamente, é apresentada uma mensagem como mostra a Figura 30.

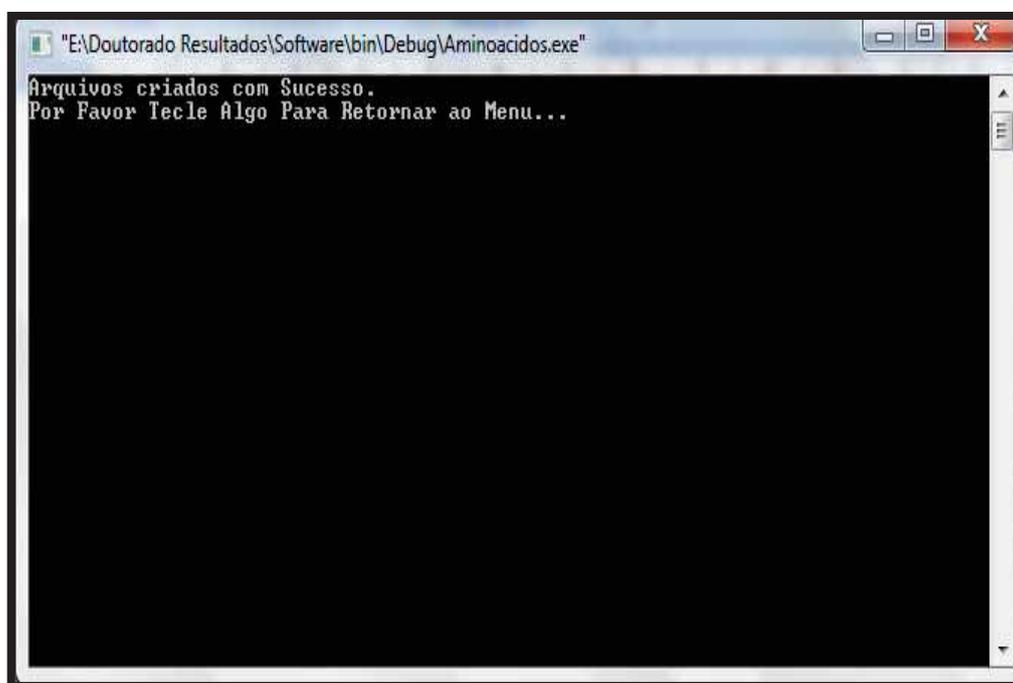
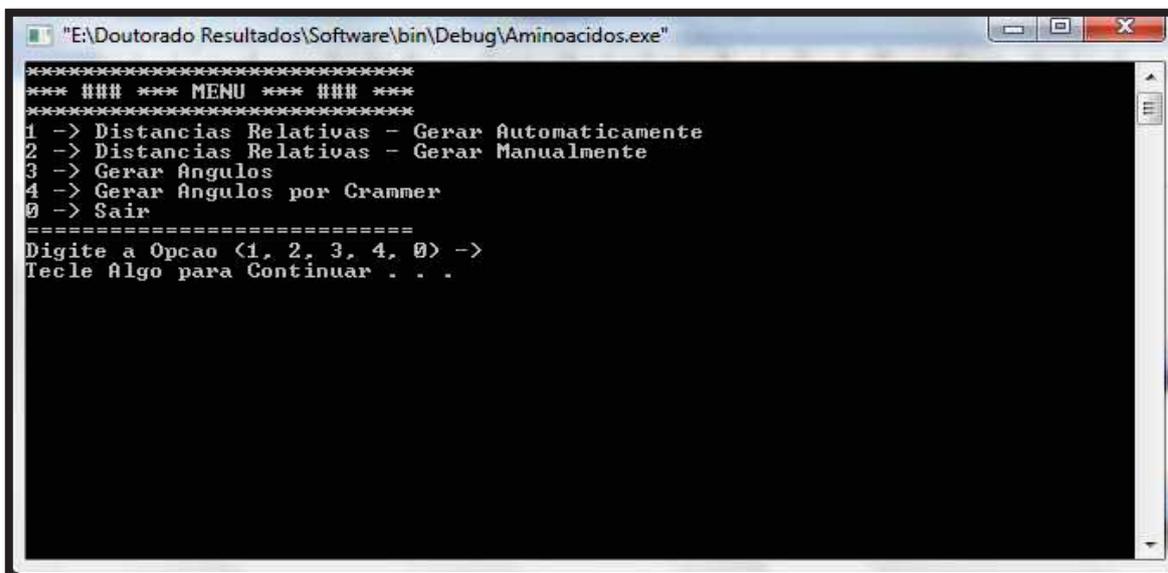


Figura 30 - Tela identificadora dos arquivos gerados corretamente. Fonte: Laboratório Simulação Molecular.

Após a execução de qualquer uma das opções, a tela inicial é mostrada para a escolha da próxima opção, conforme Figura 31.



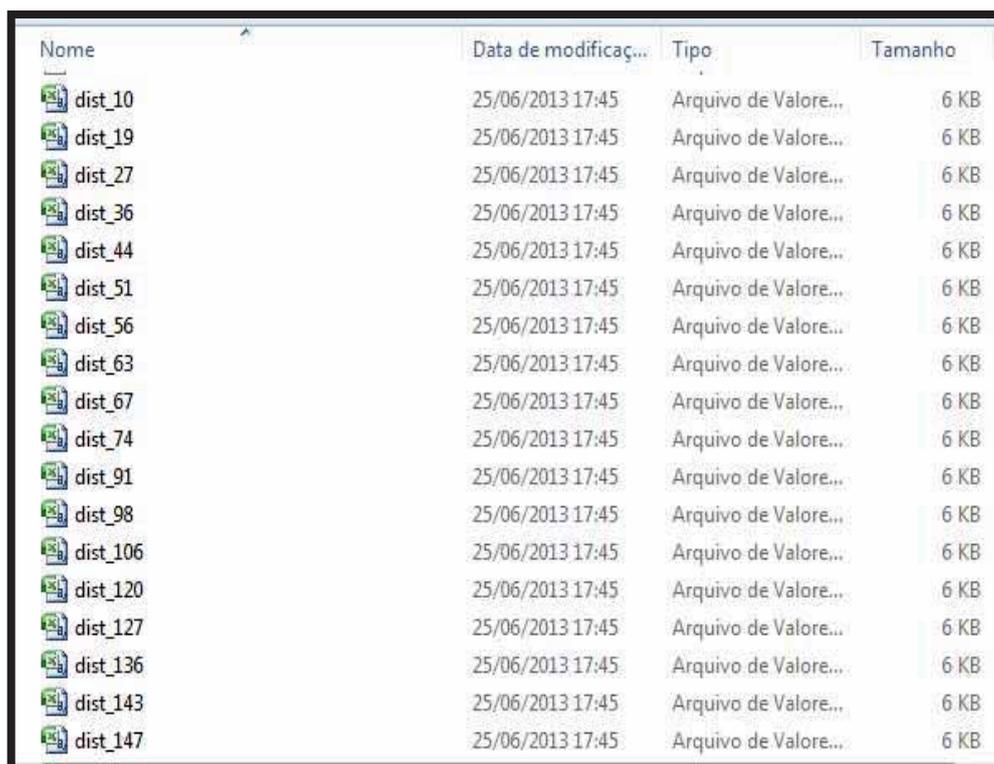
```
"E:\Doutorado Resultados\Software\bin\Debug\Aminoacidos.exe"
*****
*** ### *** MENU *** ### ***
*****
1 -> Distancias Relativas - Gerar Automaticamente
2 -> Distancias Relativas - Gerar Manualmente
3 -> Gerar Angulos
4 -> Gerar Angulos por Cramer
0 -> Sair
=====
Digite a Opcao <1, 2, 3, 4, 0> ->
Tecla Algo para Continuar . . .
```

Figura 31 - Tela inicial após qualquer opção ter sido gerada. Fonte: Laboratório Simulação Molecular.

### 3.3 Integração dos Dados

Ao realizar os cálculos da distância automática, que se encontra na opção 01, apresentam-se as distâncias de todos os carbonos alfa. Na Figura 32, estão listados alguns dos resultados da mesma estrutura, ou seja, tem-se o número de resíduos dos carbonos alfa dos quais serão calculadas as distâncias internas. Todos os arquivos são abertos no *software* Origin que facilita a integração dos dados para uma posterior análise.

Nesta etapa os dados são representados graficamente para serem analisados na próxima etapa.



Nome	Data de modificaç...	Tipo	Tamanho
dist_10	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_19	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_27	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_36	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_44	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_51	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_56	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_63	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_67	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_74	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_91	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_98	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_106	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_120	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_127	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_136	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_143	25/06/2013 17:45	Arquivo de Valore...	6 KB
dist_147	25/06/2013 17:45	Arquivo de Valore...	6 KB

Figura 32 - Resultado do cálculo da Distância automática. Fonte: Laboratório Simulação Molecular.

## 4 RESULTADOS E DISCUSSÃO

### 4.1 Resultados: Distância Interna

Nos resultados obtidos com o desenvolvimento do *software*, foram analisados os gráficos gerados por meio de implementação computacional, com o objetivo de confrontá-los com os dados da literatura e apontar alternativas para compreender a estrutura geométrica da proteína, em particular, nas regiões de Nós.

Foram selecionadas e investigadas 40 proteínas; porém, serão determinadas as distâncias internas de apenas seis delas, selecionadas aleatoriamente: 1ns5 (BENACH, J., SHEN, J., ROST, B., XIAO, R., ACTON, T., MONTELIONE, G., 2011) 2efv (KUMAREVEL, T.S., KARTHE, P., KURAMITSU, S., YOKOYAMA, S, 2011) , 1ztu (WAGNER, J.R., BRUNZELLE, J.S., FOREST, K.T., VIERSTRA, R.D, 2011), 2etl (DAS, C., HOANG, Q.Q., KREINBRING, C.A., LUCHANSKY, S.J., MERAY, R.K., RAY, S.S., LANSBURY, P.T., RINGE, D., PETSKO, G.A, 2011), 3bjx (SCHMIDBERGER, J.W., WILCE, J.A., WEIGHTMAN, A.J., WHISSTOCK, J.C., WILCE, M.C.J. 2011) e 2lyz (DIAMOND, R, 2011).

Tabela 1 - Informações das Proteínas utilizadas para analisar as distâncias internas

Código PDB	Classe	Topologia	Cadeia	Região Nó
1ns5	alfa + beta	3 <sub>1</sub>	1-153	69-121
2efv	alfa	3 <sub>1</sub>	6-87	19-66
1ztu	alfa + beta	4 <sub>1</sub>	5-325	41-298
2etl	alfa + beta	5 <sub>2</sub>	1-223	10-216
3bjx	alfa	6 <sub>1</sub>	14-296	71-268
2lyz	alfa+ beta	-	-	-

Fonte: Laboratório de Simulação Molecular (UNESP)

Os resultados aqui mostrados servem apenas para ilustrar alguns cálculos feitos com as estruturas das proteínas apontadas na Tabela 1.

Para efeito de comparação, será apresentado o resultado do cálculo de uma proteína que não possui Nó (2lyz).

A Opção 01 do *software* implementado, como foi ressaltado no item Material e Métodos, rastreia exaustivamente as distâncias internas de todas as posições da cadeia, buscando os aminoácidos carbono alfa para a realização dos cálculos. As medidas geradas para as distâncias internas estão em Å.

Conforme se observa na Figura 33, notou-se que houve pouco deslocamento em relação às distâncias internas calculadas a partir do primeiro, segundo e terceiro números de resíduo do C $\alpha$ .

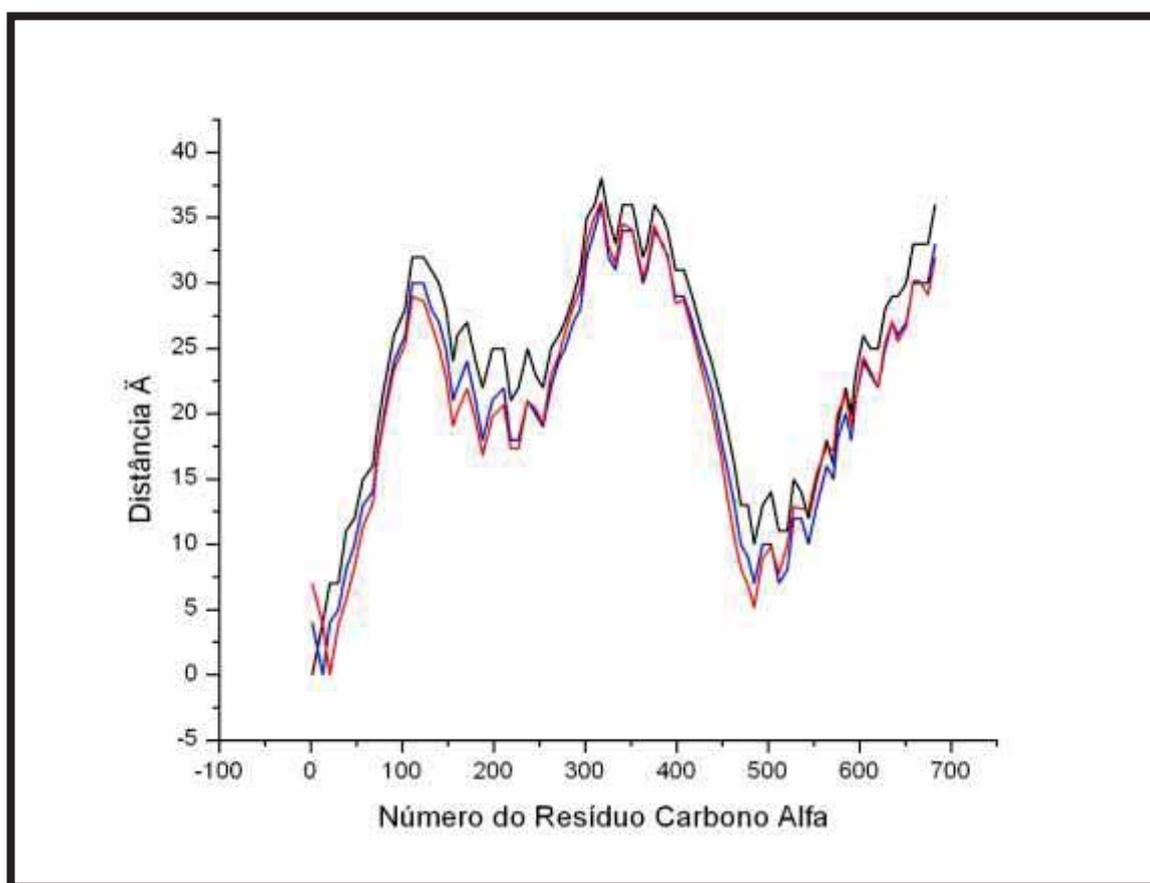


Figura 33 - Distância interna entre C $\alpha$  da proteína 2efv. A cor preta representa a primeira posição do aminoácido carbono alfa. O azul indica o cálculo de distâncias a partir da segunda posição do aminoácido carbono alfa e o vermelho é a terceira posição do carbono. Fonte: Laboratório Simulação Molecular.

A Figura 34 ilustra uma maior deformação em relação às distâncias internas de toda a proteína 2efv, o que se tornou possível devido ao fato de o cálculo iniciar na posição do meio da cadeia, ou seja, no quadragésimo terceiro aminoácido carbono alfa.

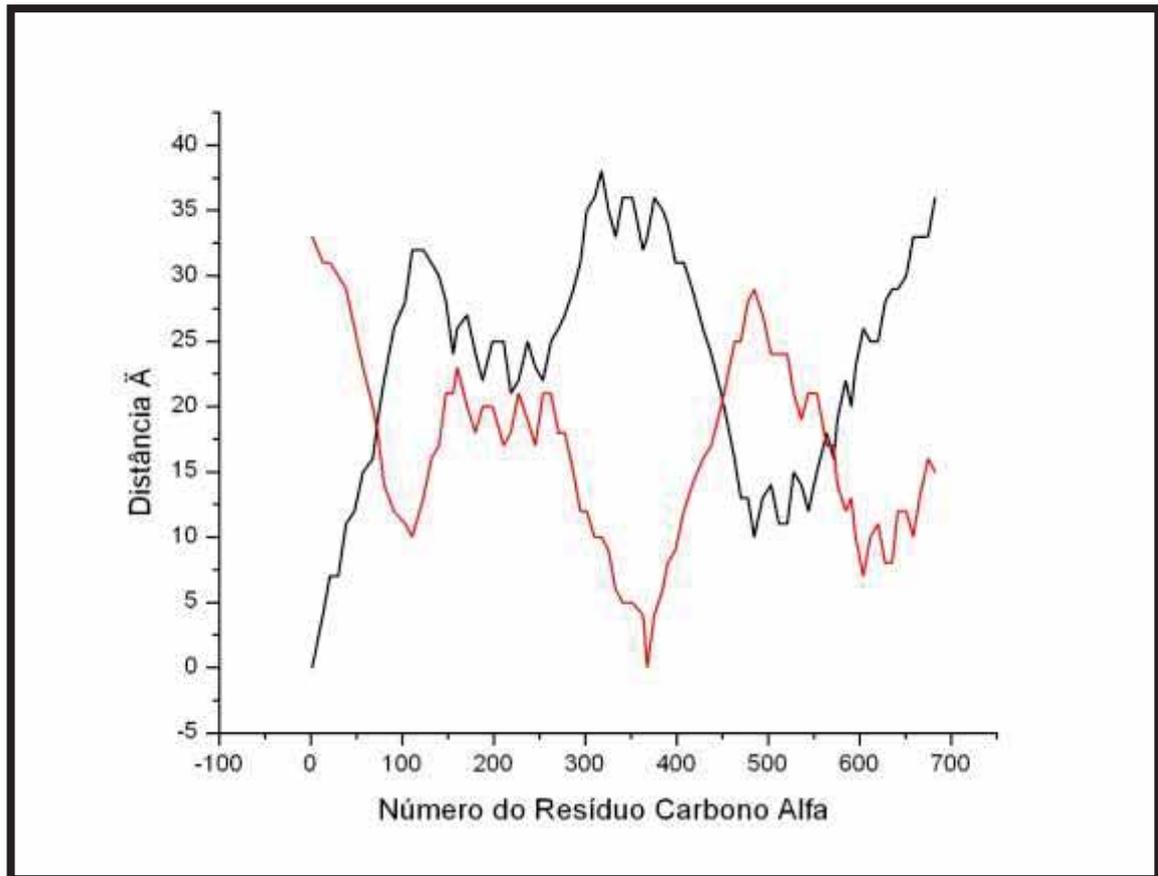


Figura 34 - Distância interna da proteína 2efv referente ao primeiro número do resíduo (em preto) e distância interna no plano do quadragésimo terceiro resíduo carbono alfa (em vermelho). Fonte: Laboratório Simulação Molecular.

Para melhor visualizar o deslocamento, apontam-se, na Figura 35, os cálculos da primeira posição do aminoácido carbono alfa em relação aos cálculos de sua última posição, verificando-se uma maior deformação.

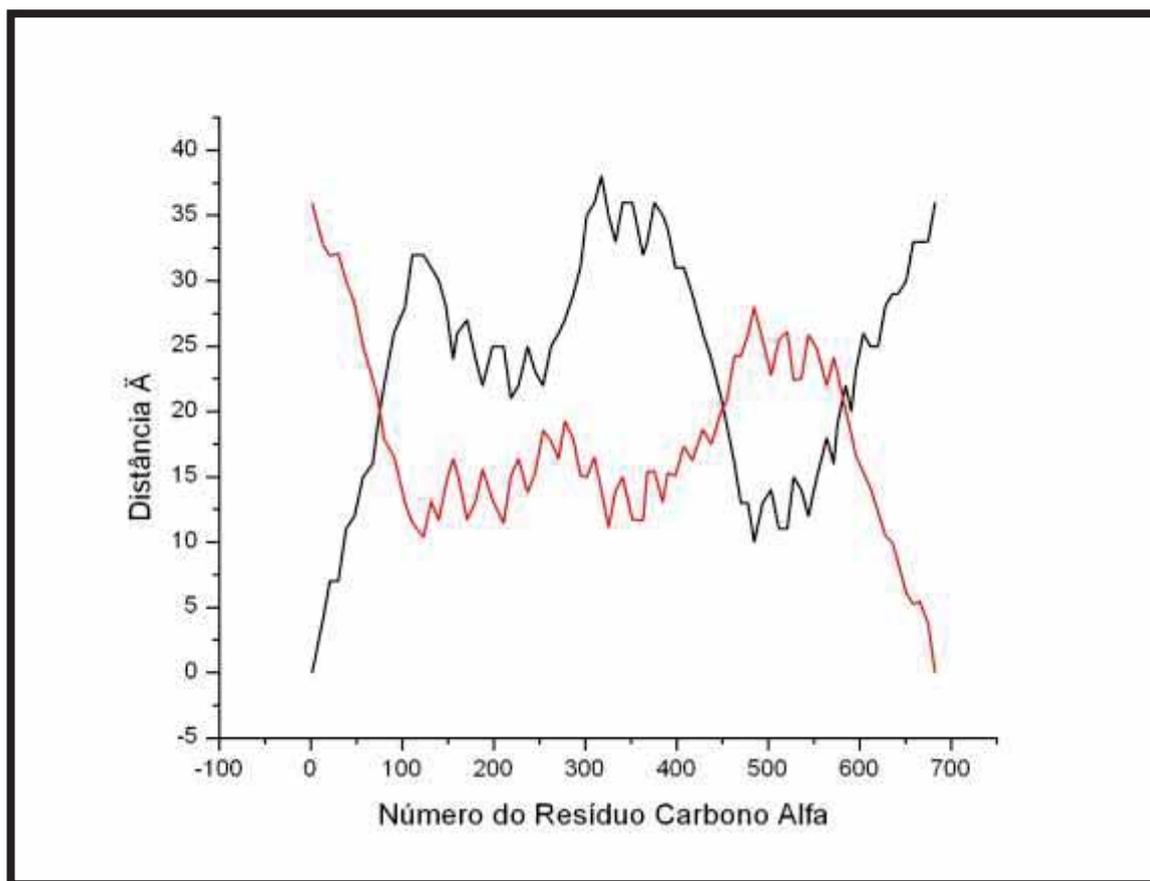


Figura 35 - Distância interna da proteína 2fv referente ao primeiro número do resíduo (em preto) e cálculo do último resíduo C $\alpha$  (em vermelho). Fonte: Laboratório Simulação Molecular.

Ao implementar essa abordagem para automação do programa de cálculo de distâncias, esperava-se reconhecer padrões visuais de comportamento na proteína através da observação dos gráficos de distância interna de todos os números dos resíduos C $\alpha$  referentes à posição diferente da cadeia.

Ocorreu o rastreamento de todos os resíduos do C $\alpha$  e calcularam-se as distâncias internas de seus números, a fim de analisar o comportamento dos carbonos  $\alpha$  no plano e entender a sua forma geométrica.

Constatou-se que, nos cálculos de distância dos primeiros resíduos, há pouca deformação. Contudo, no meio da cadeia, observou-se um deslocamento maior que possibilitou melhor visualização dos aminoácidos C $\alpha$ . Entretanto, ao

compará-lo com os cálculos de distâncias do último número de resíduo, verificou-se uma maior deformação, a qual impossibilitou a leitura do gráfico, isto é, da distância entre o C $\alpha$  do primeiro resíduo e o C $\alpha$  do n-esimo resíduo.

Diante disso, subentendeu-se que os cálculos de distância do n-esimo resíduo são mais abertos levando à perda de sua estrutura original.

Concluiu-se, portanto, que o resultado de distância interna contribui para analisar o comportamento, no plano, de todos os resíduos C $\alpha$ , em todas as posições possíveis da cadeia, porém, não contempla o reconhecimento de padrões em proteínas, o que compreenderia a primeira abordagem.

#### **4.1.1 Resultados: Distância Interna Proteína 1ns5**

Os resultados na Figura 36 referem-se à estrutura da proteína código PDB 1ns5, cuja topologia matemática é  $3_1$ . Esta proteína possui duas cadeias, sendo que, para cada uma delas, se verificam as distintas regiões dos Nós. Tal Figura é gerada pelo servidor web para a confirmação dos Nós e de sua região de formação, resultando em uma representação gráfica simplificada da estrutura.

Resultados das Distâncias Internas da Proteína 1ns5				
Knot residues	Chain start-stop	Knot type	Knot	
69-121A	1-153A	$\underline{3}_1$ (trefoil knot)		<a href="#">Jmol visualization</a>
69-116B	1-155B	$\underline{3}_1$ (trefoil knot)		<a href="#">Jmol visualization</a>

**Residues 69-121A**

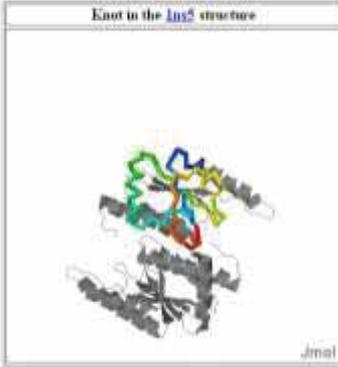
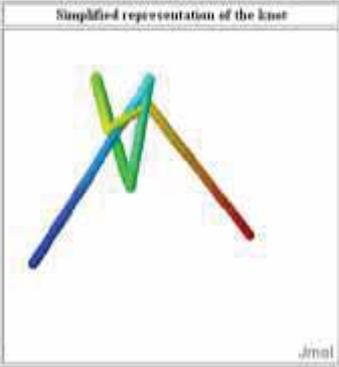
Knot in the 1ns5 structure	Simplified representation of the knot
	

Figura 36 - Proteína 1ns5. Fonte: <http://knots.mit.edu/>.

Após a análise pelo servidor web, a Figura 37 apresenta os cálculos de distâncias internas da cadeia A. A região em vermelho representa a demarcação da área do Nó matemático. O mesmo gráfico foi gerado para a região da cadeia B e obteve-se o mesmo resultado.

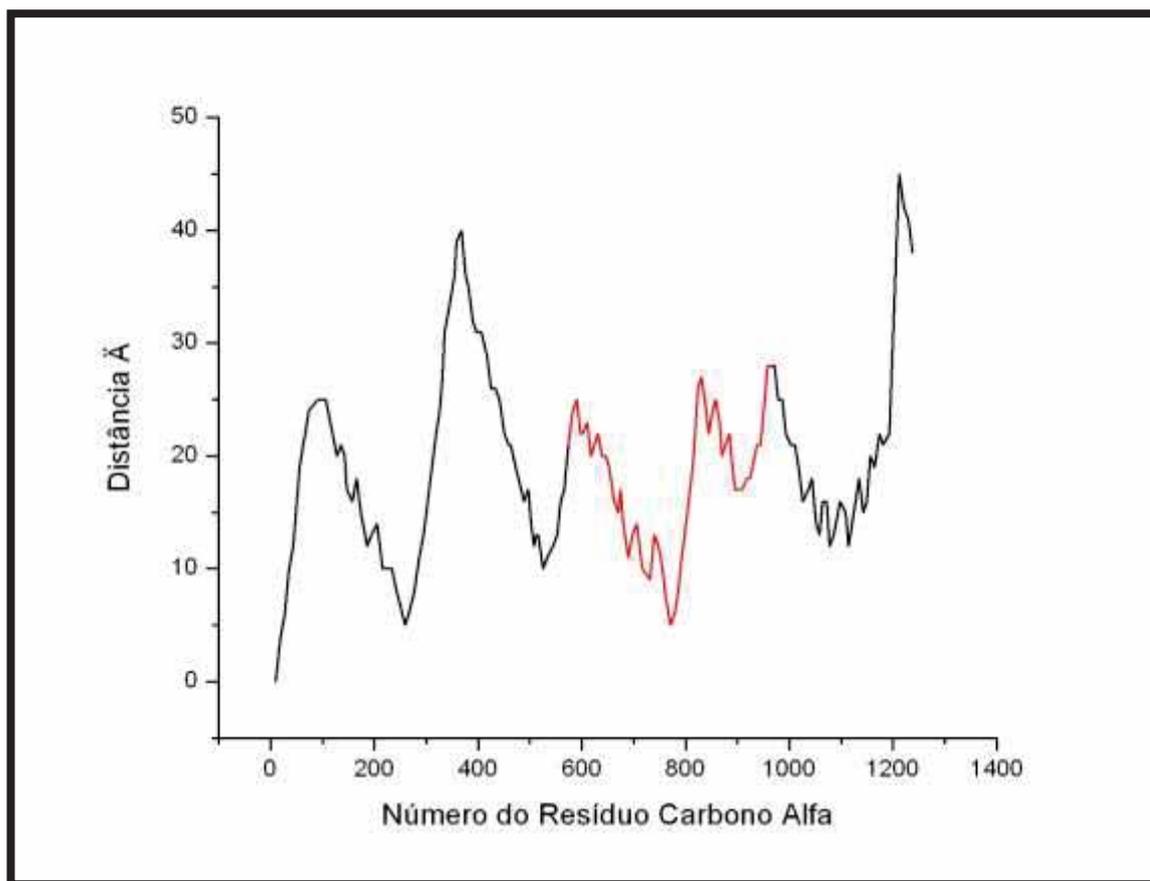


Figura 37 - Distância interna da proteína 1ns5 cadeia A. Em vermelho, a demarcação da região do Nó. Fonte: Laboratório Simulação Molecular.

Verificou-se uma deformação do Nó da proteína em relação ao Nó da matemática, pois a topologia  $3_1$  apresenta uma simetria que dificilmente será encontrada na região do Nó de uma proteína real.

Identificaram-se as distâncias internas do Nó matemático  $3_1$  e constatou-se a sua simetria, conforme Figura 38.

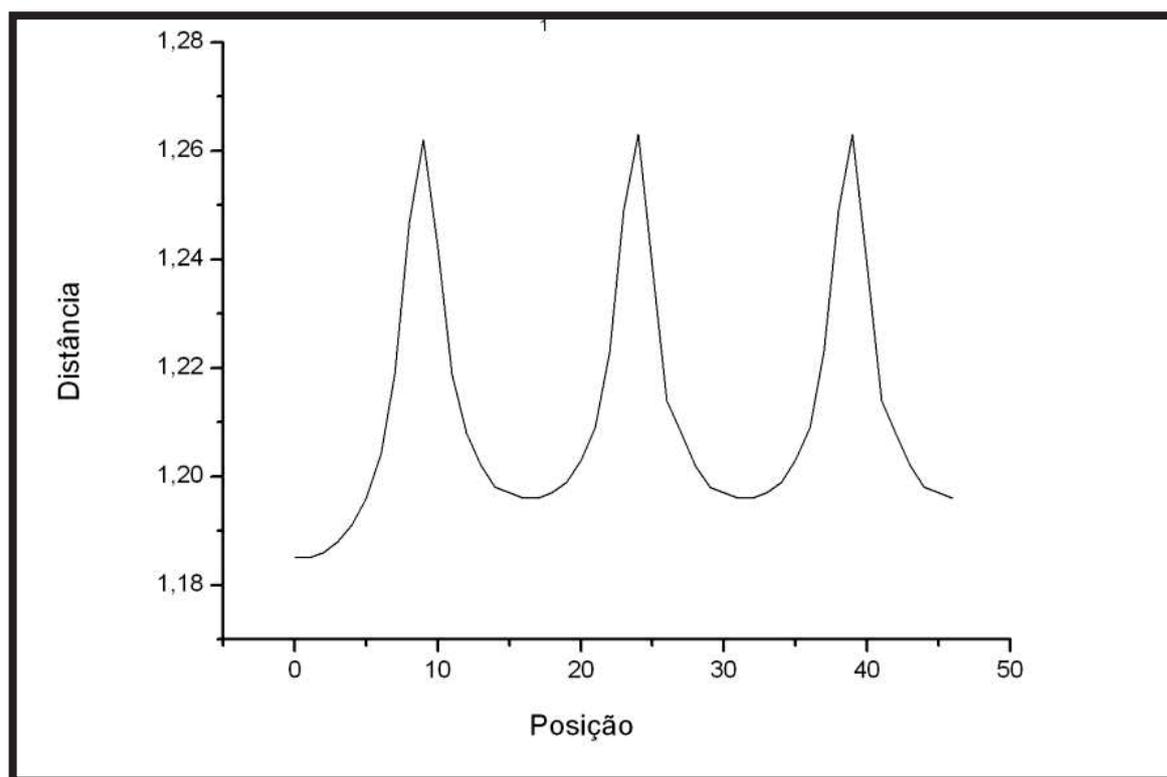


Figura 38 - Distância interna do Nó Matemático 31 unidades arbitrárias.  
Fonte: Laboratório Simulação Molecular.

Devido ao grau de deformação visto na Figura 37, destacado em vermelho, pode-se notar a quantidade de concavidades e picos que formam o Nó matemático e observar a similaridade existente. Todavia, sem a demarcação da região do Nó, como visto na literatura, torna-se impossível a detecção do mesmo pelo gráfico de distância interna.

#### 4.1.2 Resultados: Distância Interna Proteína 2efv

Por meio do servidor web, verificou-se a confirmação dos Nós, sua região de formação e a simplificação gráfica de sua estrutura. A proteína é pequena, possui uma cadeia e sua topologia matemática é o Nó  $3_1$ .

Na Figura 39, apresentam-se as distâncias internas da proteína. A demarcação em vermelho identifica os carbonos alfa que compõem a região do Nó.

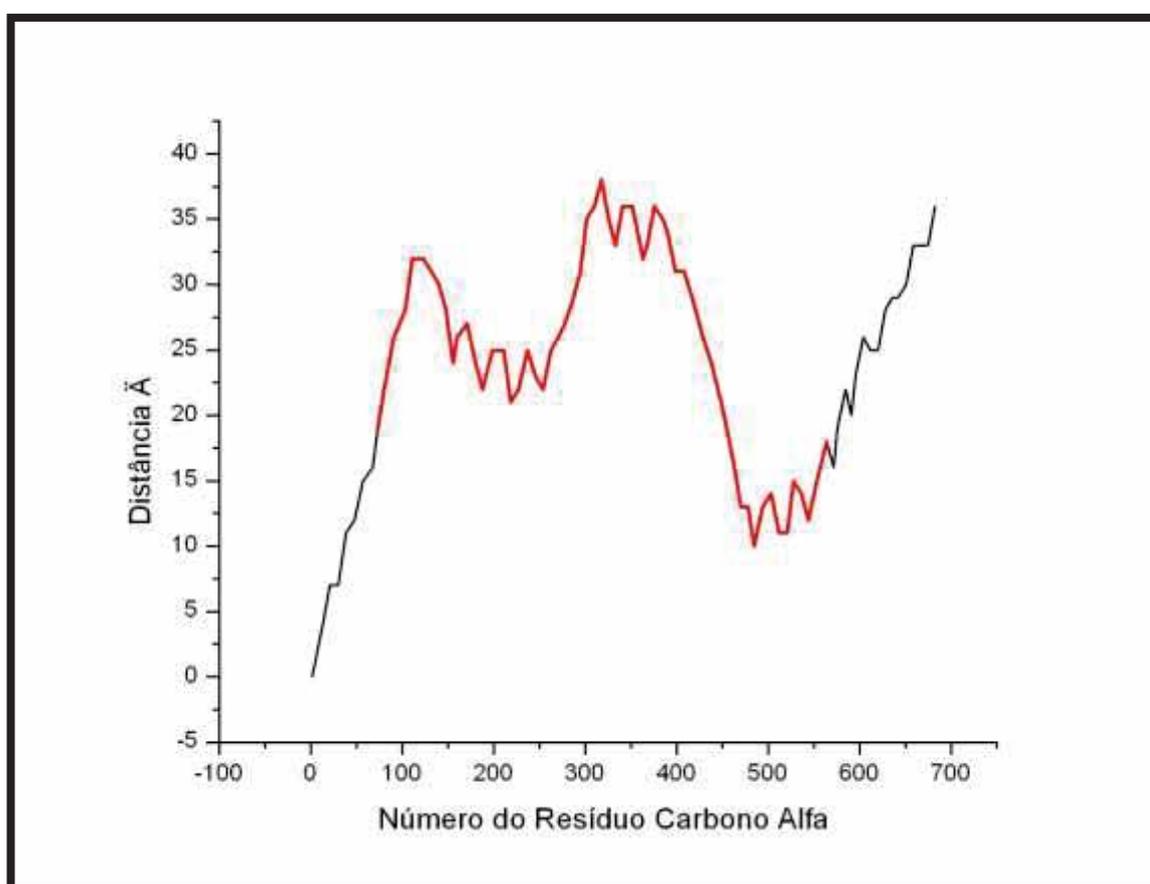


Figura 39 - Distância interna da proteína 2efv. Em vermelho, a demarcação da região do Nó.  
Fonte: Laboratório Simulação Molecular.

Após verificar-se o comportamento dos Nós no plano, a mesma região foi projetada em 3D. Desse modo, observou-se que os resíduos dos C $\alpha$  não realizam o cruzamento referente ao Nó matemático, como mostra a Figura 40.

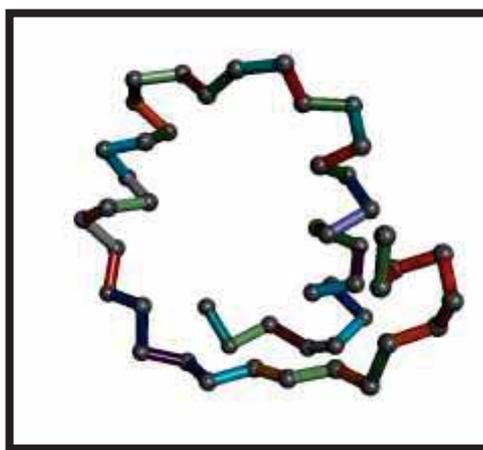


Figura 40 -Distâncias internas dos carbonos alfa da região do Nó, conforme literatura (VIRNAU, P.; MIRNY, A. L.; KARDAR, M., 2006).

Sugeriu-se, então, que fossem ampliados os resíduos que compõem o Nó. Assim, pode-se ter a confirmação visual dos movimentos que pertencem à formação do Nó matemático.

A proposta é que sejam eliminados do C-terminal e N-terminal apenas cinco carbonos alfa de cada lado para que se possa oferecer uma descrição mais adequada às características do Nó matemático, como foi feito na Figura 41.

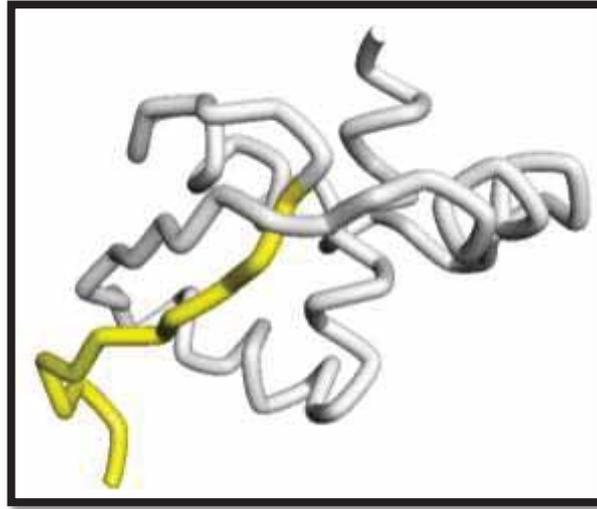


Figura 41 - Distâncias internas retirando cinco carbonos alfas de cada lado.  
Fonte: Laboratório Simulação Molecular.

#### 4.1.3 Resultados: Distância Interna Proteína 1ztu

Como se pode verificar na Figura 42, gerada pelo servidor web para a confirmação dos Nós e de sua região de formação, bem como para a simplificação gráfica da sua estrutura, a proteína 1ztu possui um domínio. O Nó matemático referente a essa proteína é a topologia  $4_1$ .

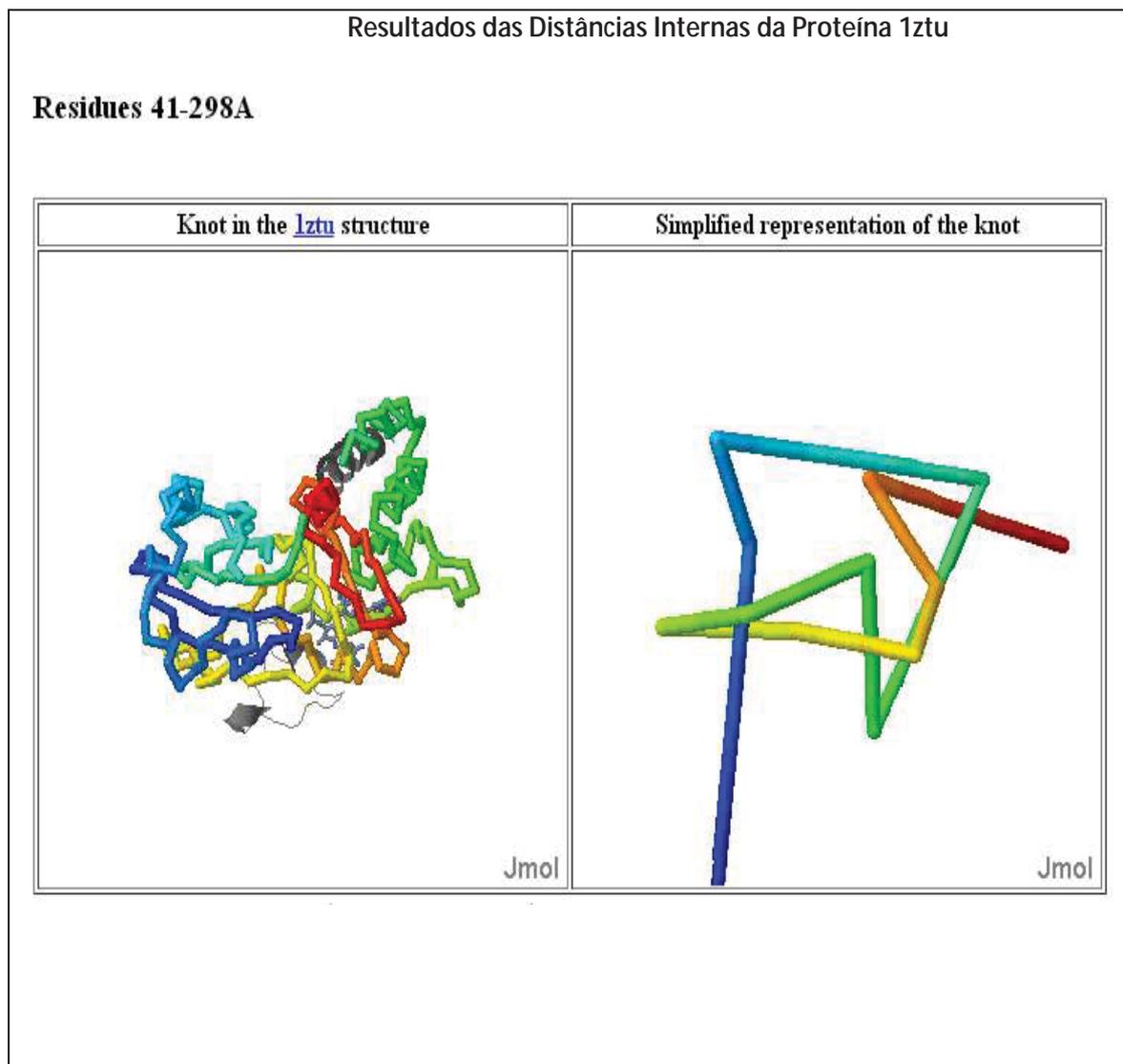


Figura 42 - Proteína 1ztu. Fonte: <http://knots.mit.edu/>.

Nota-se que o grau de complexidade aumenta com topologias acima de três cruzamentos. Na Figura 43, são apontadas as distâncias internas da proteína 1ztu, sendo a região demarcada em vermelho aquela referente à região do Nó matemático.

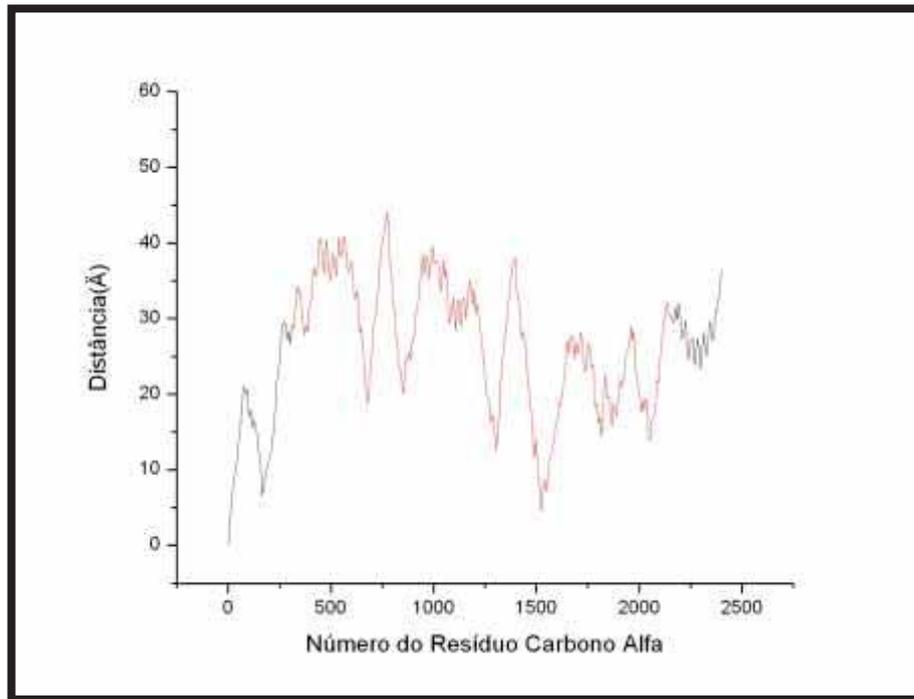


Figura 43 - Distância interna Proteína 1ztu. Demarcação em vermelho referente à topologia do Nó  $4_1$ .  
Fonte: Laboratório Simulação Molecular.

Constatou-se, assim, a dificuldade de identificação do Nó matemático por similaridade. A topologia matemática do Nó  $4_1$  ilustra as suas respectivas distâncias internas.

Conclui-se, portanto, que, para o Nó matemático  $4_1$ , não houve uma visualização adequada que possibilitasse a identificação da região do Nó como esperávamos. Mesmo embasados na literatura, foi impossível detectar qualquer similaridade, conforme a Figura 44.

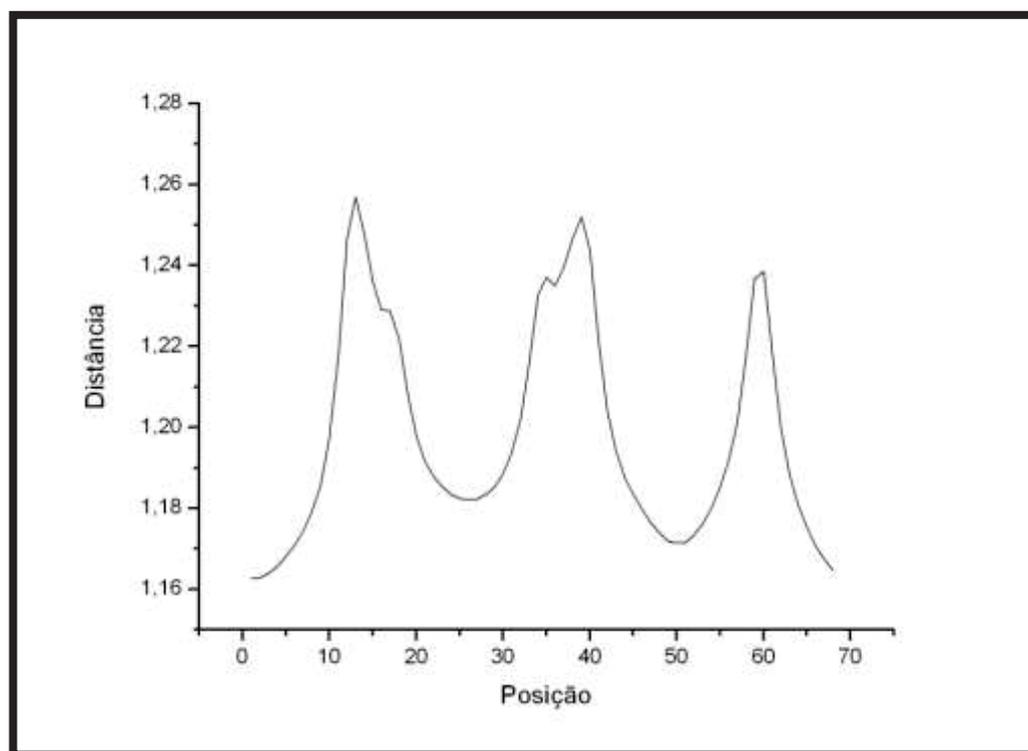


Figura 44 - Distância interna do Nó matemático 4<sub>1</sub>. Fonte: Laboratório Simulação Molecular.

#### 4.1.4 Resultados: Distância Interna Proteína 2etl

Na Figura 45, pode-se observar a proteína 2etl gerada pelo servidor web para a confirmação dos Nós, de sua região de formação, para a simplificação gráfica de sua estrutura e a ilustração de duas cadeias.

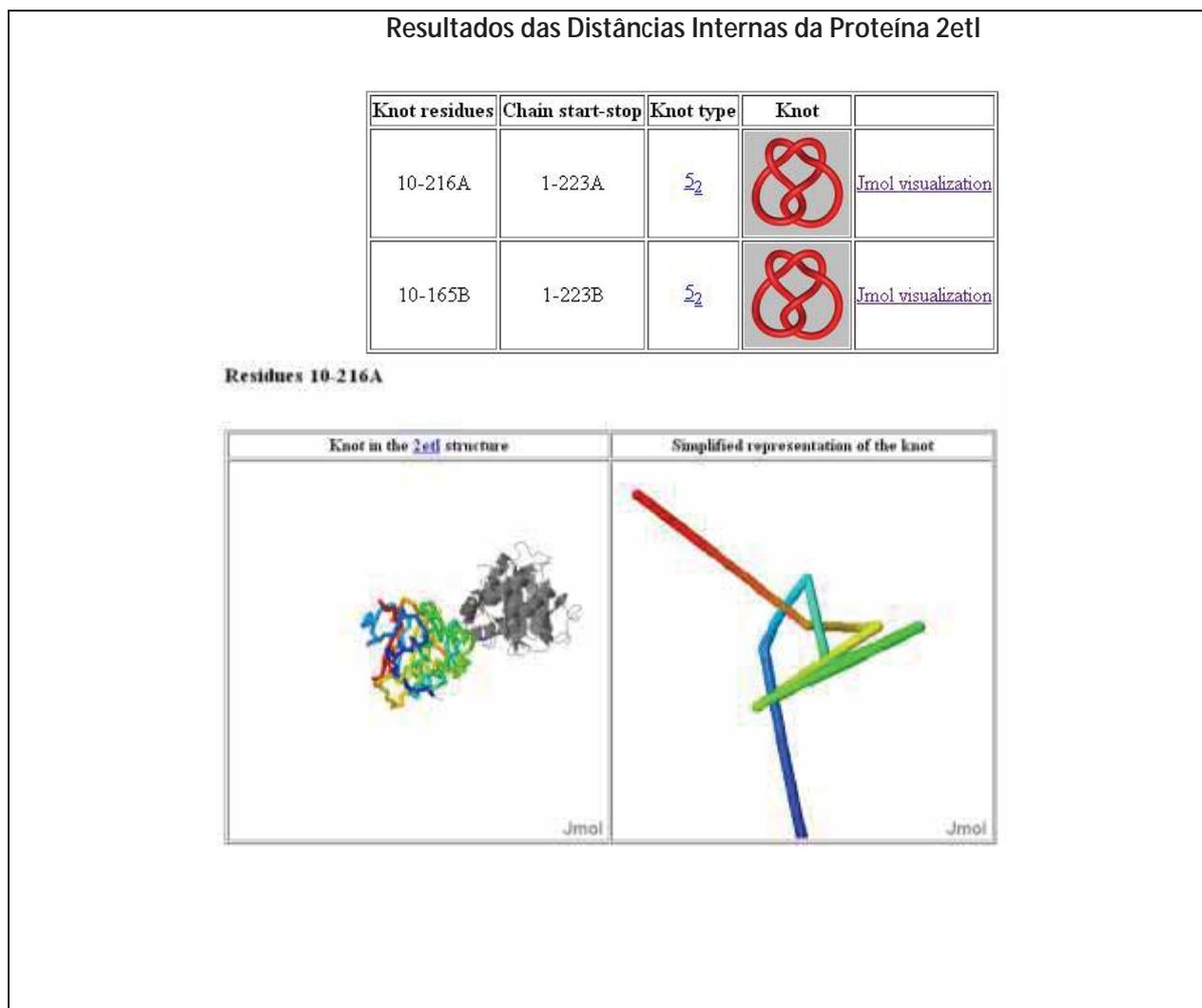


Figura 45 - Proteína 2etl. Fonte: <http://knots.mit.edu/>.

Percebe-se que o resultado apresentado para a topologia  $5_2$  não difere da topologia  $4_1$ , resultante de um grau de complexidade para análise de gráficos de distâncias internas com mais de três cruzamentos. O Nó matemático  $5_2$  pode ser visto na Figura 46.

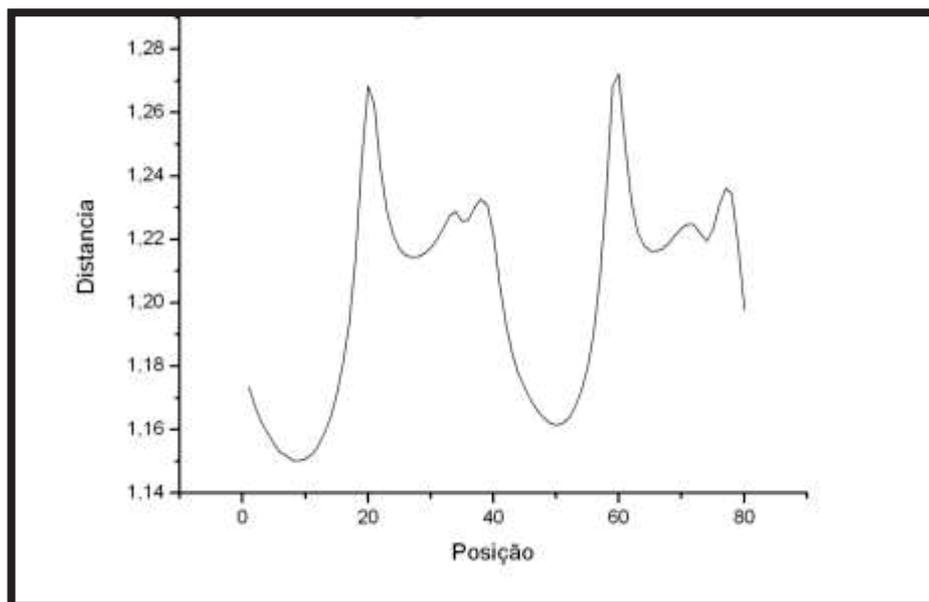


Figura 46 - Distância interna topologia matemática 52 unidades arbitrárias.  
Fonte: Laboratório Simulação Molecular.

Notou-se a distância entre os dois gráficos apresentados referente às distâncias internas da proteína 5<sub>2</sub>.

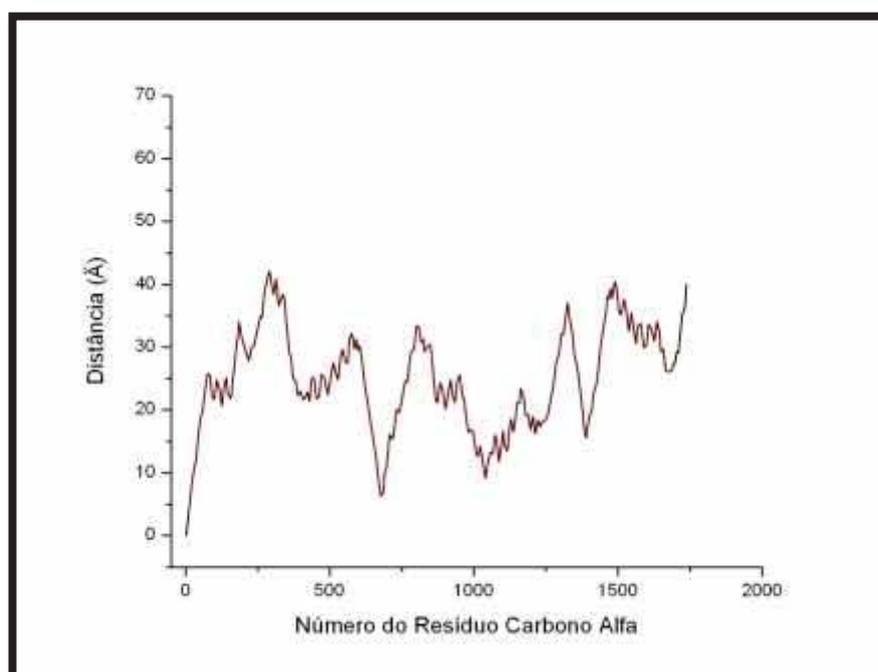


Figura 47 - Distância interna da cadeia A. Em vermelho, a região do Nó 5<sub>2</sub>.  
Fonte: Laboratório Simulação Molecular.

#### 4.1.5 Resultados: Distância Interna Proteína 3bjx

A estrutura da Proteína 3bjx, apresentada na Figura 48, possui quatro cadeias. Esta figura foi gerada pelo servidor web para a confirmação dos Nós e de sua região de formação e para a simplificação gráfica de sua estrutura.

O ponto de interrogação gerado pelo servidor web significa que não foi possível detectar o Nó 6<sub>1</sub>. A topologia do Nó pode ser afirmada mediante consulta à literatura (VIRNAU, P.; MIRNY, A. L.; KARDAR, M., 2006).

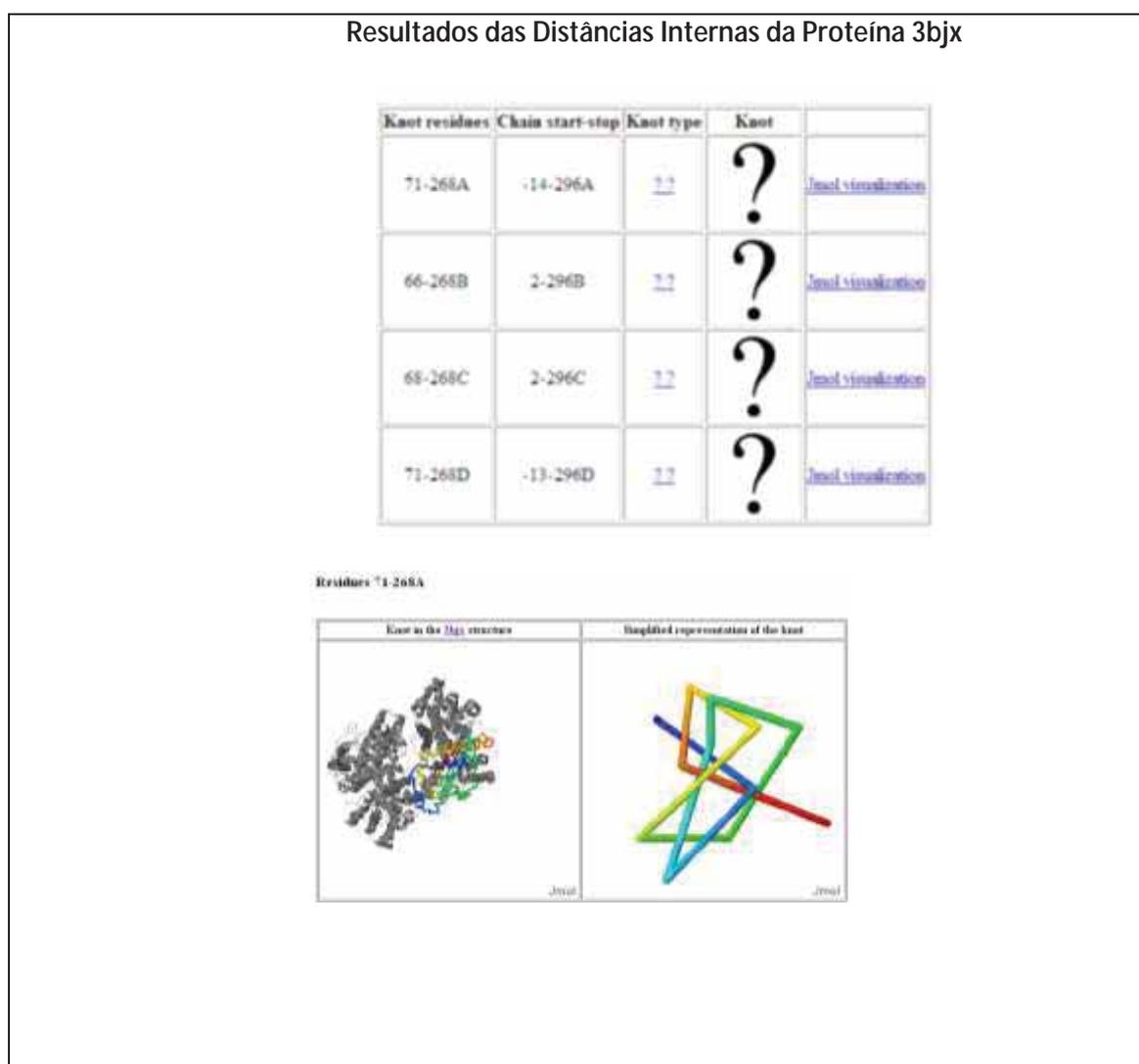


Figura 48 - Proteína 3bjx. Fonte: <http://knots.mit.edu/>.

Em relação ao resultado da distância interna para o Nó 6<sub>1</sub> e a proteína que o apresenta, não houve concordância. Ressalta-se, ainda, que a dificuldade de observação no plano se intensifica na medida em que o número de cruzamentos aumenta.

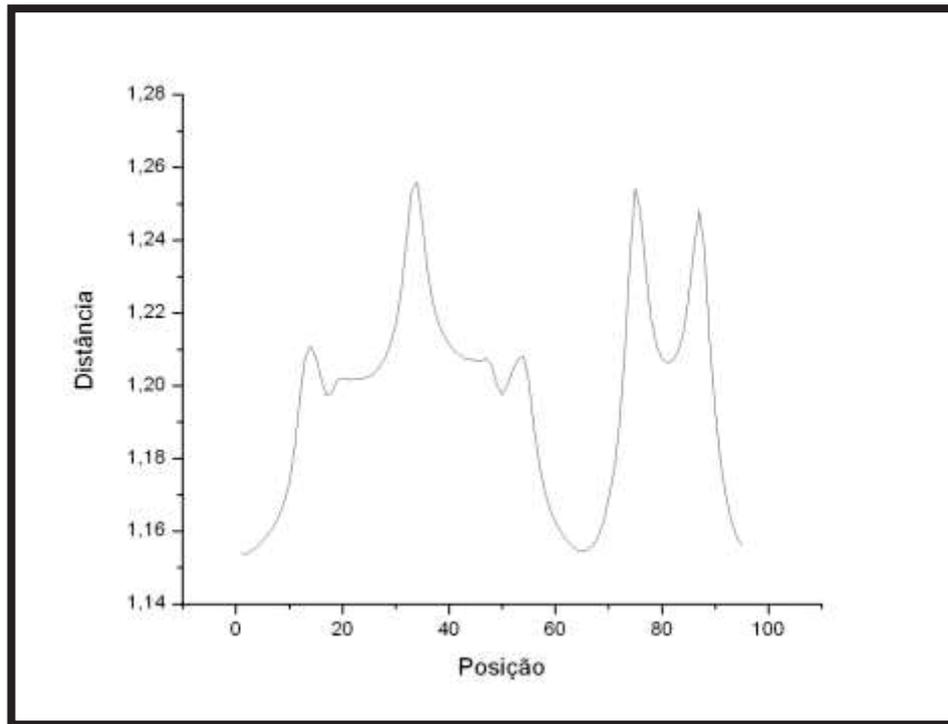


Figura 49 - Distância interna da topologia matemática Nó 6<sub>1</sub>. Fonte: Laboratório Simulação Molecular.

A partir da literatura, identificaram-se as regiões dos Nós na proteína 3bjx, evidenciando o seu grau de complexidade para a análise.

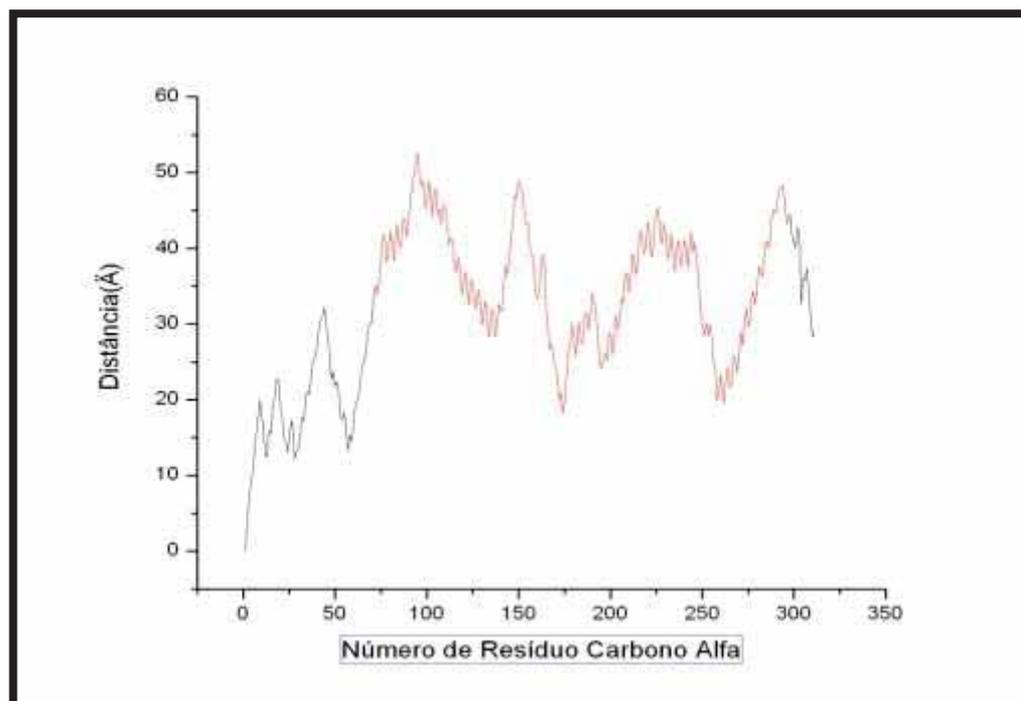
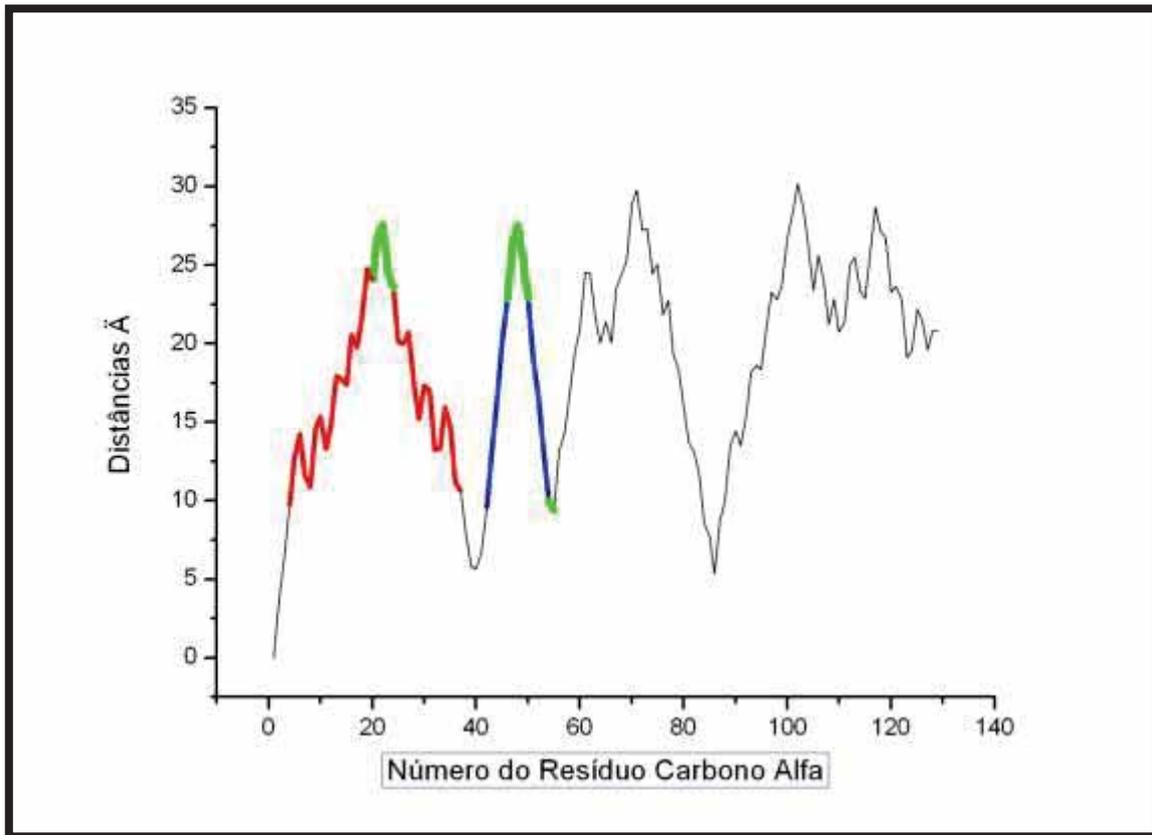


Figura 50 -Distância interna proteína 3bjx. Em vermelho, região do Nó 6. Fonte: Laboratório Simulação Molecular.

#### 4.1.6 Resultados: Distância Interna Proteína 2lyz

O resultado referente à simulação da proteína Lizozima 2lyz que não possui Nó pode ser visualizado na Figura 51.



**Figura 51 - Distâncias internas da Lisozima 2lyz.** Fonte: Laboratório Simulação Molecular.

Considera-se impossível assegurar se a Figura 51 possui ou não Nó matemático, uma vez que os Nós não apresentaram padrão de comportamento que pudesse ser identificado pela análise gráfica. É com base na literatura (VIRNAU, P.; MIRNY, A. L.; KARDAR, M., 2006) e no servidor web que afirmo, portanto, que a Lisozima não possui Nó matemático.

Sendo assim, uma maneira encontrada para visualizar as estruturas secundárias foi através do cálculo de distâncias internas dos carbonos alfa. Todavia, esta identificação somente se torna possível com o diagrama do BPBSum, pois, como se ressaltou anteriormente, apenas pela observação do gráfico não se pode fazer tal afirmação.

Na Figura 52, as regiões destacadas em vermelho representam a hélice  $\alpha$ . Já as demarcações em azul indicam a folha  $\beta$ . Os turns, por sua vez, foram assinalados em verde na Figura 51.

A Figura 52 traz o diagrama da proteína que está localizado no PDBSum. Nela, mostra-se a correlação com a Figura 51 apontando os aminoácidos que possuem hélice  $\alpha$ , folha  $\beta$  e turns.

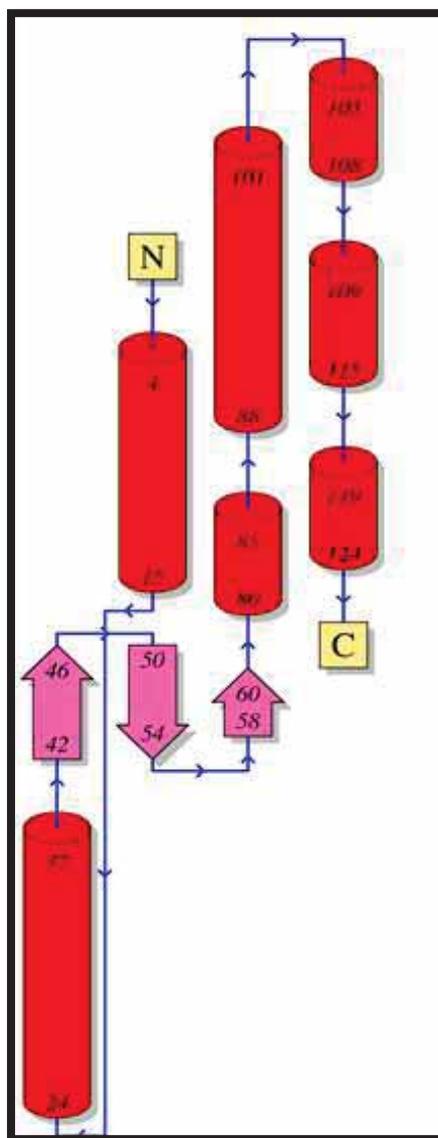


Figura 52 - PDBSum

## 4.2 Resultados: Diedros

Os resultados mostrados a seguir foram gerados utilizando-se do programa desenvolvido no laboratório de simulação molecular.

A Figura 53 apresenta o ângulo diedro da proteína 2efv. Na figura, o eixo das abscissas representa a posição para os cálculos referentes ao ângulo diedro gerado entre os planos compostos pela posição dos aminoácidos. Já o eixo das ordenadas corresponde aos resultados dos diedros calculados.

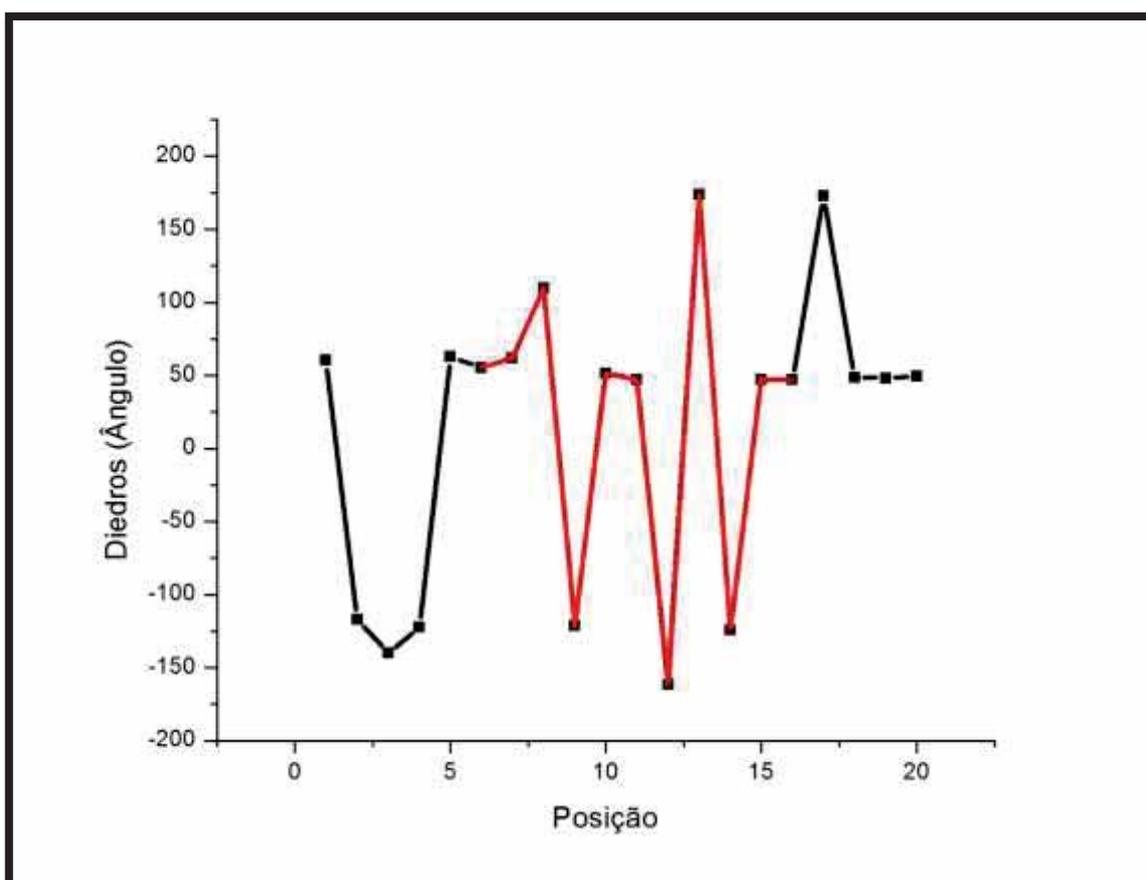


Figura 53 - Diedros de todos os carbonos alfa. Em vermelho, Diedro da região do Nó. Fonte: Laboratório Simulação Molecular.

Observou-se que os ângulos diedros gerados no plano, como mostra a Figura 53, foram baseados apenas nos planos compostos por carbono alfa. Para melhor visualização, a Tabela 2 traz o modo como foram gerados os planos para o cálculo de diedros. Elaborou-se esta tabela com a finalidade de averiguar como os planos foram calculados no *software* desenvolvido no laboratório de simulação. A primeira coluna é a de ordenação; a segunda introduz os carbonos alfa para a formação do plano e a terceira apresenta os resultados dos cálculos gerados pelo *software*.

TABELA 2 - Planos para cálculos dos diedros

Posição	Número dos Carbonos Alfas envolvidos nos cálculos	Diedros
1	Plano 1 - ATOM 2,13,21 / Plano 2 – ATOM 13,21,30	60,35
2	Plano 1 - ATOM 39,48,57 / Plano 2 – ATOM 48,57,68	-116,76
3	Plano 1 - ATOM 73,80,91 / Plano 2 – ATOM 80,91,103	-140,160
4	Plano 1 - ATOM 111,123,132 / Plano 2 – ATOM 123, 132,140	-122,20
5	Plano 1 - ATOM 148,156,160 / Plano 2 – ATOM 156,160,171	62,93
6	Plano 1 - ATOM 180,188,199 / Plano 2 – ATOM 188,199,211	55,31
7	Plano 1 - ATOM 219,227,237 / Plano 2 – ATOM 227, 237,246	62,02
8	Plano 1 - ATOM 254,263,271/ Plano 2 – ATOM 263,271,278	109,66
9	Plano 1 - ATOM 287,295,302 / Plano 2 – ATOM 295,302,310	-121,16
10	Plano 1 - ATOM 318,326,333 / Plano 2 – ATOM 326,333,341	51,63
11	Plano 1 - ATOM 352,363,368 / Plano 2 – ATOM 363,368,376	47,34
12	Plano 1 - ATOM 385,390,399 / Plano 2 – ATOM 390,399,408,417	-161,51
13	Plano 1 - ATOM 417,429,438 / Plano 2 – ATOM 429,438,449	173,73
14	Plano 1 - ATOM 455,463,470 / Plano 2 – ATOM 463,470,478	-124,14
15	Plano 1 - ATOM 485,494,503 / Plano 2 – ATOM 494,503,512	47,01
16	Plano 1 - ATOM 521,528,536 / Plano 2 – ATOM 528,536,544	47,23
17	Plano 1 - ATOM 553,564,572 / Plano 2 – ATOM 564,572,576	172,72
18	Plano 1 - ATOM 585,591,596 / Plano 2 – ATOM 591,596,604	48,73
19	Plano 1 - ATOM 612,620,628 / Plano 2 – ATOM 620,628,636	48,29
20	Plano 1 - ATOM 642,651,659 / Plano 2 – ATOM 651,659,666	49,55

Fonte: Laboratório de Simulação Molecular (UNESP)

A Figura 54 ilustra o que se denomina de posição dos diedros no gráfico. Observa-se que dois planos geram um diedro, cujo resultado está atrelado a uma posição no plano cartesiano. A proteína 2efv possui 87 carbonos alfa e 20 posições que identificam os respectivos diedros.

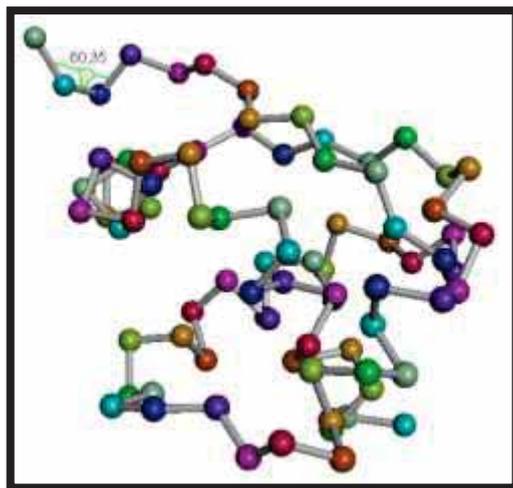


Figura 54 - Posição 1 Proteína 2efv. Fonte: Laboratório Simulação Molecular.

Na Figura 55, apresentam-se os diedros da Lisozima 2lyz. Especifica-se que esta proteína não possui Nó. Os diedros são exibidos no plano com o objetivo de visualizar a conformação de uma proteína, uma vez que o enovelamento desta depende dos diedros.

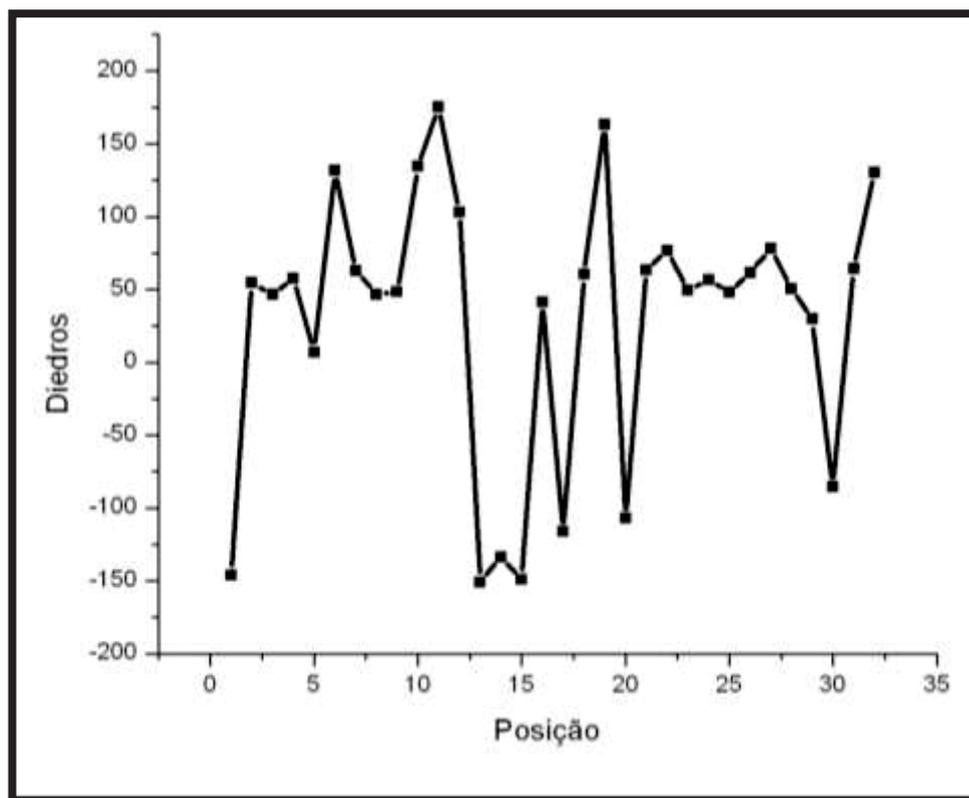


Figura 55 - Diedros da Lisozima. Fonte: Laboratório Simulação Molecular.

A Lisozima contém 32 ângulos diedros entre os carbonos alfa, os quais contribuem para estabilizar a conformação de sua estrutura. Observe a Tabela 3:

TABELA 3 - Mostra os cálculos dos diedros da proteína 2lyz

Posição	Diedros
1	-146,02
2	54,74
3	46,52
4	57,8
5	7,18
6	131,75
7	62,99

---

<b>8</b>	46,83
<b>9</b>	48,45
<b>10</b>	134,46
<b>11</b>	175,38
<b>12</b>	103,2
<b>13</b>	-150,75
<b>14</b>	-133,68
<b>15</b>	-148,91
<b>16</b>	41,67
<b>17</b>	-116,11
<b>18</b>	60,39
<b>19</b>	163,45
<b>20</b>	-107,1
<b>21</b>	63,43
<b>22</b>	76,98
<b>23</b>	49,6
<b>24</b>	56,96
<b>25</b>	48,05
<b>26</b>	61,68
<b>27</b>	78,05
<b>28</b>	50,74
<b>29</b>	29,77
<b>30</b>	-85,64
<b>31</b>	64,23
<b>32</b>	130,31

---

Fonte: Laboratório de Simulação Molecular (UNESP)

A hipótese inicial em relação aos cálculos dos diedros versava sobre a busca por padrões, de modo que se pudesse fazer alguma correlação com a área do

Nó. No entanto, não houve esta constatação. Os resultados encontrados contribuíram para a visualização do comportamento no plano do enovelamento.

### 4.3 Resultados Distância ponta-ponta proteína inteira.

A distância ponta-ponta consiste em um dos fatores importantes que mantém a estrutura tridimensional. Esta distância é representada em Å.

TABELA 4 - Distância Ponta-Ponta

Código	Tamanho	Distância
PDB		Ponta-Ponta
1ns5	153	37.657
2efv	87	35.946
1flj	259	49.964
1jd0	260	29.284
1kop	223	29.434
1rj6	259	43.067
1y7w	274	19.398
1znc	262	23.419
1hcb	258	41.367
1keq	238	19.976
1lug	259	37.635
1v9e	259	40.044
1z93	263	52.186

Fonte: Laboratório de Simulação Molecular (UNESP)

A Figura 56 ilustra a distância ponta-ponta de uma proteína.

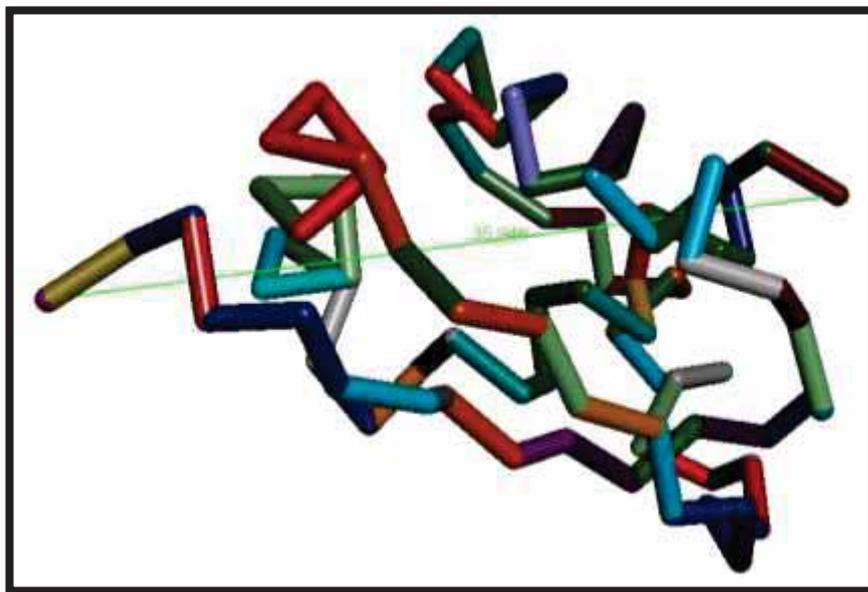


Figura 56 - Proteína 2efv e seus respectivos carbonos alfa indicando, em sua cadeia, a distância ponta-ponta. Fonte: Laboratório Simulação Molecular.

#### 4.4 Resultados Distância ponta-ponta região do Nó

TABELA 5 - Distância ponta-ponta – região do Nó

Código	Tamanho	Região do	Distância
PDB		Nó	Ponta-Ponta
1ns5	153	69-121	15.043
2efv	87	19-66	19.807
1flj	259	30-256	9.898
1jd0	260	28-257	20.780
1kop	223	36-223	10.323
1rj6	259	29-257	14.441
1y7w	274	37-270	12.810
1znc	262	32-261	14.882
1hcb	258	29-256	14.962
1keq	238	7-234	14.401
1lug	259	30-256	14.519
1v9e	259	32-256	10.237
1z93	263	28-254	14.670

Fonte: Laboratório de Simulação Molecular (UNESP)

A Figura 57 ilustra a distância ponta-ponta de uma proteína em sua região do Nó.

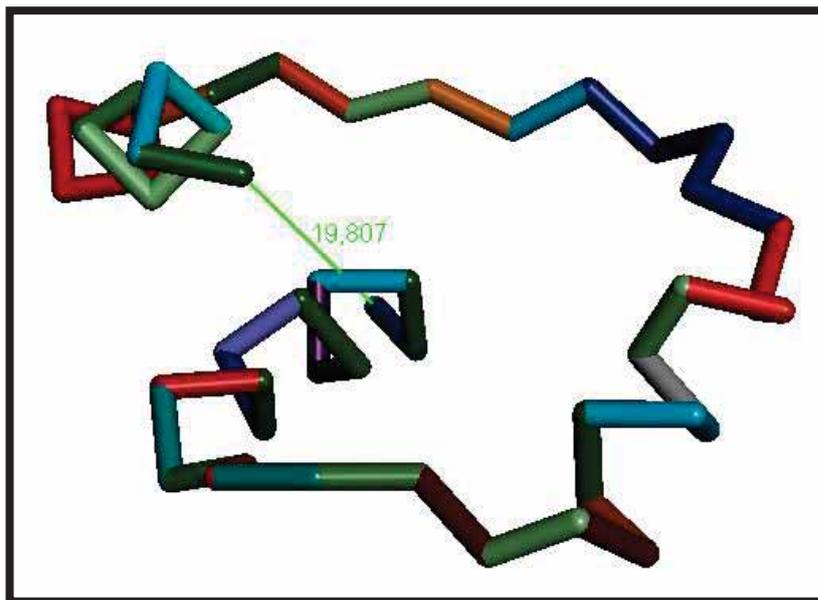


Figura 57 - Proteína 2efv e seus respectivos carbonos alfa indicando a distância ponta-ponta em sua configuração apenas da região do Nó. Fonte: Laboratório Simulação Molecular.

Em relação aos resultados de distância ponta-ponta, percebe-se que, na região do Nó, demarcada na Figura 56, não há o Nó tridimensional. Em razão disso, os resultados apresentados fundamentam-se na literatura (VIRNAU, P.; MIRNY, A. L.; KARDAR, M., 2006).

Observou-se, também, que a distância ponta-ponta da região do Nó é menor do que a distância inteira da proteína, o que se acredita decorrer do fato de o Nó estar frequentemente no interior da cadeia e possuir maior estabilidade.

#### 4.5 Escala de Hidrofobicidade Região do Nó (*Knot*)

Os aminoácidos com cadeias laterais polares são chamados de hidrofílicos, ou seja, têm afinidade com a água; já os aminoácidos com cadeias laterais apolares são denominados hidrofóbicos, isto é, repelem a água.

A Tabela 6 apresenta a escala de hidrofobicidade utilizada neste trabalho bem como a escala de Kyte-Doolittle usada para delinear características hidrofóbicas de uma proteína.

Tabela 6 - Escala Kyte-Doolittle

Arg	-4,5
Lys	-3,9
Asp	-3,5
Asp Acid	-3,5
Glut	-3,5
Glut Acid	-3,5
Hist	-3,2
Prol	-1,6
Tyro	-1,3
Trypt	-0,9
Ser	-0,8
Threo	-0,7
Gly	-0,4
Ala	1,8
Meth	1,9
Cyst	2,5
Pheny	2,8
Leu	3,8
Val	4,2
Iso	4,5

Fonte: J. Mol. Biol. 157:105-132(1982)

A sequência da proteína 2efv comparece na Tabela 7. Na primeira coluna, encontram-se os aminoácidos; na segunda, a sua posição; e, na terceira, a sua hidrofobicidade.

Tabela 7 - Hidrofobicidade da região do nó proteína 2efv

Aminoácido	Posição	Hidrofobicidade
Arginine	1	-4,5
Lysine	2	-3,9
Leucine	3	3,8
Arginine	4	-4,5
Tyrosine	5	-1,3
Leucine	6	3,8
Leucine	7	3,8
Histidine	8	-3,2
Lysine	9	-3,9
Leucine	10	3,8
GlutamicAcid	11	-3,5
Asparagine	12	-3,5
Valine	13	4,2
GlutamicAcid	14	-3,5
Asparagine	15	-3,5
Valine	16	4,2
AsparticAcid	17	-3,5
Isoleucine	18	4,5
AsparticAcid	19	-3,5
Threonine	20	-0,7
Leucine	21	3,8
Arginine	22	-4,5
Arginine	23	-4,5
Alanine	24	1,8
Isoleucine	25	4,5
GlutamicAcid	26	-3,5
Alanine	27	1,8
GlutamicAcid	28	-3,5
Lysine	29	-3,9
Lysine	30	-3,9
Tyrosine	31	-1,3
Lysine	32	-3,9
Arginine	33	-4,5
Serine	34	-0,8
Isoleucine	35	4,5
Threonine	36	-0,7
Leucine	37	3,8
Threonine	38	-0,7
GlutamicAcid	39	-3,5
GlutamicAcid	40	-3,5
GlutamicAcid	41	-3,5
GlutamicAcid	42	-3,5
Valine	43	4,2
Isoleucine	44	4,5
Isoleucine	45	4,5
Glutamine	46	-3,5
Arginine	47	-4,5
Leucine	48	3,8

Fonte: PDB, 2013

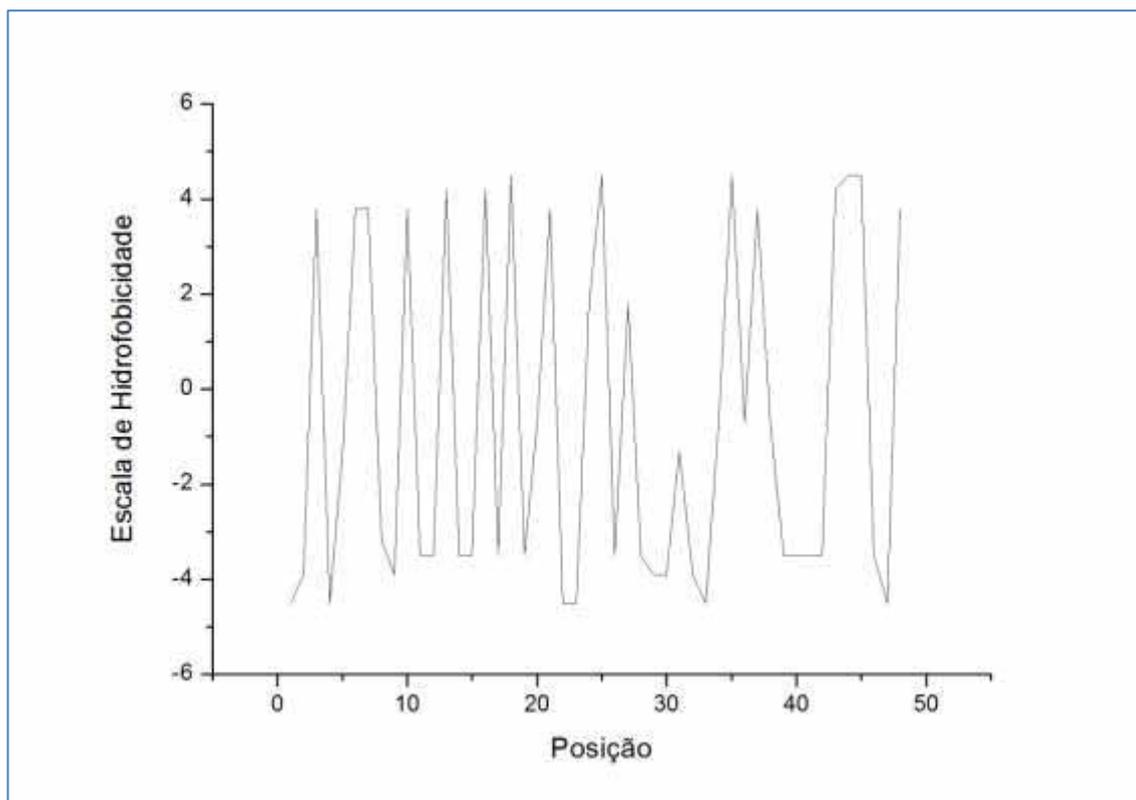


Figura 58 - Gráfico Hidrofobicidade em relação aos aminoácidos 2efv. Fonte: Laboratório Simulação Molecular.

A Figura 58 representa graficamente a Tabela 7 onde se percebeu aminoácidos com maior hidrofobicidade na região do Nó. Não houve o resultado esperado uma vez que a hipótese inicial era que se pudesse encontrar maior hidrofobicidade na região do Nó.

#### 4.6 Frequência de aminoácidos hidrofóbicos

A Tabela 8 exibe as quantidades de ocorrência dos mesmos aminoácidos na região do Nó.

Tabela 8 - Frequência de Aminoácidos da Proteína 2efv – Região do Nó (Knots)

Aminoácido	Número de Aminoácidos	Hidrofóbico/ Hidrofílico
Arginine	6	Hidrofílico
Lysine	5	Hidrofílico
Leucine	7	Hidrofóbico
Tyrosine	2	Hidrofóbico
Histidine	1	Hidrofílico
GlutamicAcid	8	Hidrofílico
Asparagine	2	Hidrofílico
Valine	3	Hidrofóbico
Isoleucine	5	Hidrofóbico
AsparticAcid	2	Hidrofílico
Threonine	3	Hidrofílico
Alanine	2	Hidrofóbico
Serine	1	Hidrofílico
Glutamine	1	Hidrofílico
Total	48	

Fonte: Laboratório de Simulação Molecular (UNESP)

A representação gráfica se encontra na Figura 59, apresentada a seguir. Nela, para a região do Nó na proteína 2efv, verificou-se a maior incidência de ácido glutâmico existente na escala de hidrofobicidade Kyte-Doolittle -3,5.

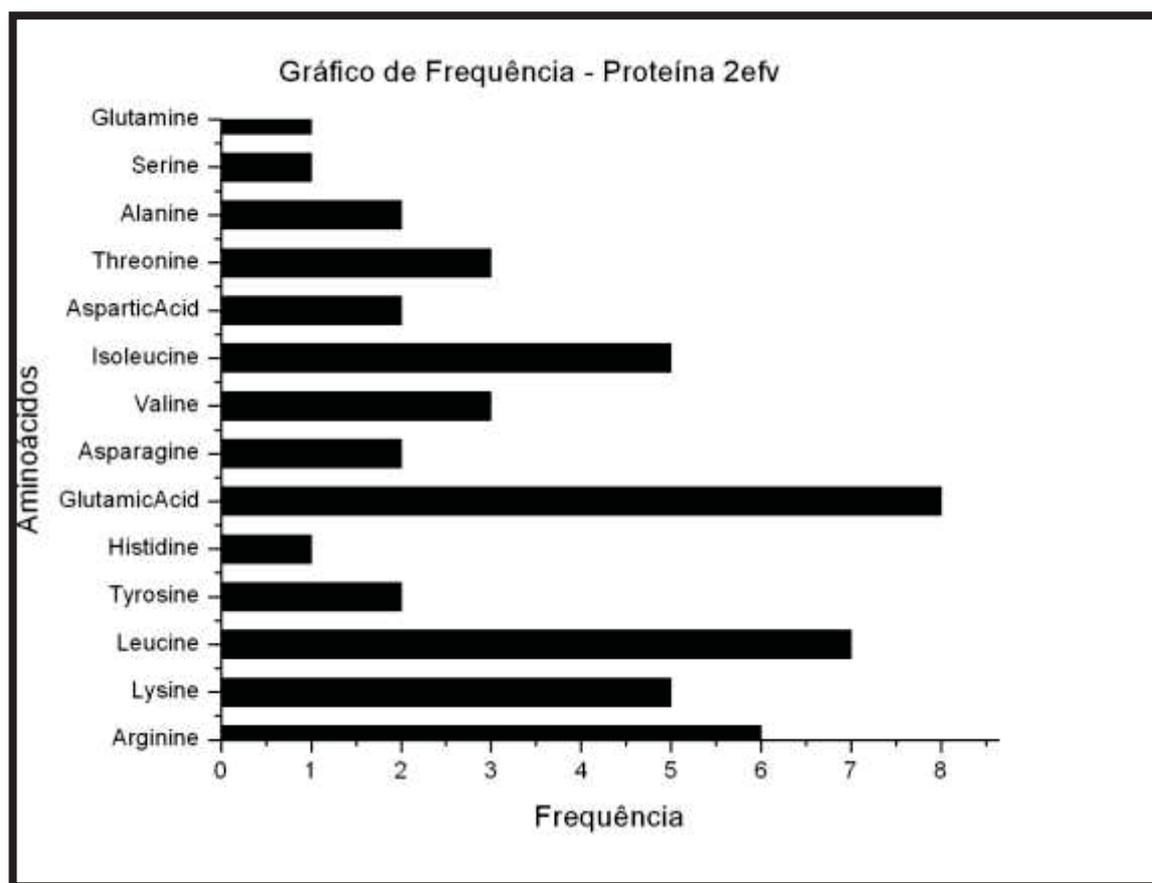


Figura 59 - Frequência da Proteína 2efv região do Nó. Fonte: Laboratório Simulação Molecular.

Nos resultados apresentados em relação à escala de hidrofobicidade, verificou-se maior incidência de hidrofílicos, refutando a hipótese inicial segundo a qual haveria maior incidência de hidrofóbicos na região do Nó, visto que o seu posicionamento estava no interior da cadeia.

A Tabela 9 mostra o levantamento de incidências dos aminoácidos na proteína inteira e a Figura 60, a sua representação gráfica. A análise dos dados indicou a incidência do aminoácido Leucina e Alanina, com maior frequência, entretanto, de aminoácidos hidrofílicos

Tabela 9 - Frequência de Aminoácidos da Proteína 1ns5

Aminoácidos	Número de Aminoácidos	Hidrofóbico/ Hidrofílico
SelenoMethionine	4	Hidrofóbico
Lysine	11	Hidrofílico
Leucine	19	Hidrofóbico
Glutamine	5	Hidrofílico
Valine	8	Hidrofóbico
Alanine	15	Hidrofóbico
Glycine	11	Hidrofílico
Threonine	8	Hidrofílico
Proline	12	Hidrofílico
AsparticAcid	8	Hidrofílico
Tryptophan	5	Hidrofílico
Phenylalanine	3	Hidrofóbico
GlutamicAcid	10	Hidrofílico
Tyrosine	3	Hidrofílico
Arginine	9	Hidrofílico
Isoleucine	8	Hidrofóbico
Asparagine	3	Hidrofílico
Histidine	3	Hidrofílico
Serine	7	Hidrofílico
Cysteine	1	Hidrofílico
Total	153	

Fonte: Laboratório de Simulação Molecular (UNESP)

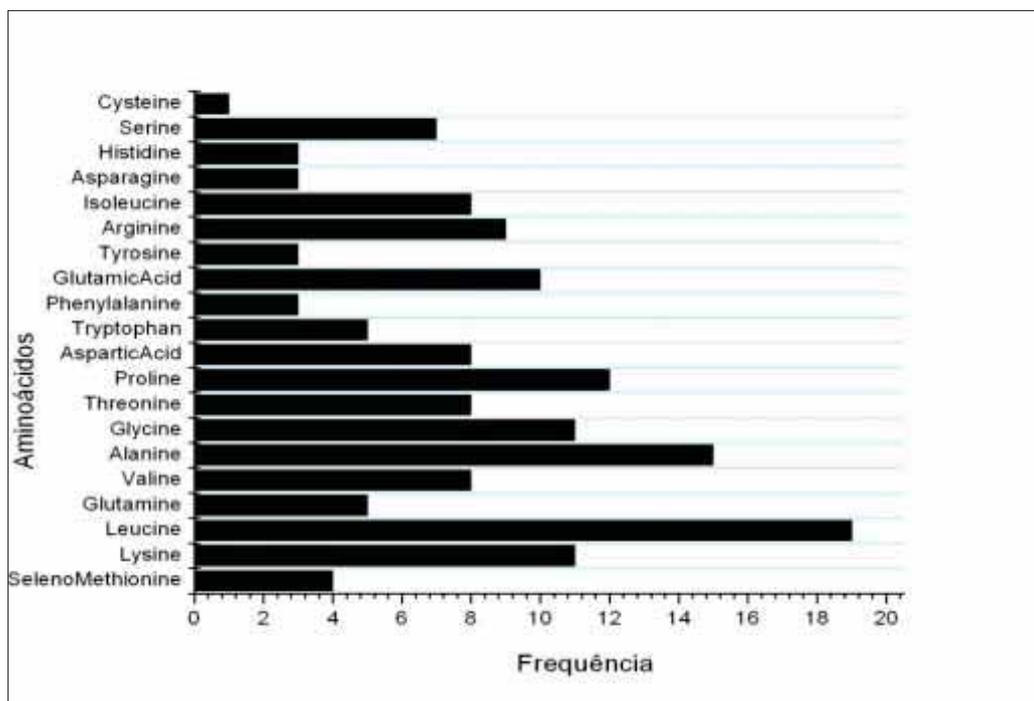


Figura 60 - Frequência de Aminoácidos. Fonte: Laboratório Simulação Molecular.

Assim, em relação às frequências dos aminoácidos, concluiu-se que não há uma relação de hidrofobicidade dos  $C\alpha$  como se esperava.

## 5 CONCLUSÕES E CONSIDERAÇÕES FINAIS

Nos gráficos de distâncias internas, encontrou-se uma abordagem para identificar hélice  $\alpha$ , folha  $\beta$  e turns. Entretanto, para a realização do mesmo, há, ainda, a necessidade de parâmetros estabelecidos na literatura.

Houve uma razoável concordância referente às proteínas em sua distância interna na topologia matemática, a  $3_1$ . Foi possível detectar similaridade em relação a quantidades de concavidades e picos em proteína real. Porém, faz-se necessário o aporte da literatura para a confirmação da área dos Nós.

Apresentou-se o comportamento do enovelamento pelos diedros do  $C\alpha$  no plano assim como uma abordagem para conhecer a descrição de seu caminho neste. Todavia, tal abordagem não contempla o encontro de padrões na região do Nó.

Constatou-se que as distâncias ponta-ponta da região dos Nós são menores devido ao fato de sua posição estar no interior da cadeia.

Houve uma discordância com a região apresentada na literatura referente aos  $C\alpha$  que compõem o Nó e, por essa razão, sugeriu-se uma ampliação destes para que, na projeção 3D, se possa visualizar o emaranhado e os movimentos que caracterizam o Nó.

Verificou-se, ainda, pela escala de hidrofobicidade, que os aminoácidos são mais hidrofílicos na região do Nó, contrariando a hipótese inicial.

Em vista do exposto, concluiu-se que os objetivos propostos no trabalho foram bem desenvolvidos. Contudo, não houve validação do programa para reconhecimento de padrões em Nós em proteínas reais.

Para estudos futuros será descrito o comportamento das topologias matemáticas apresentadas em forma de equações, o que possibilitará a descrição de proteínas que possuam Nós matemáticos parametrizados.

A equação paramétrica de uma curva é a sua representação através de equações que expressam as coordenadas dos pontos da curva como funções de uma variável chamada parâmetro. Um exemplo é o círculo unitário descrito pela equação  $x^2 + y^2 = 1$ . Esta equação pode ser parametrizada tendo  $t$  como parâmetro.

$$[\cos(t), \sin(t)], 0 \leq t \leq 2\pi \quad \text{Equação 18}$$

Outras equações paramétricas mais sofisticadas são possíveis, como, por exemplo, a hélice, cuja equação é dada abaixo (na notação do GeoGebra 5):

$$[2\cos(3t), 2\sin(3t), t, 0, 100] \quad \text{Equação 19}$$

Na Figura 61, temos a representação desta equação paramétrica no *software* GeoGebra.

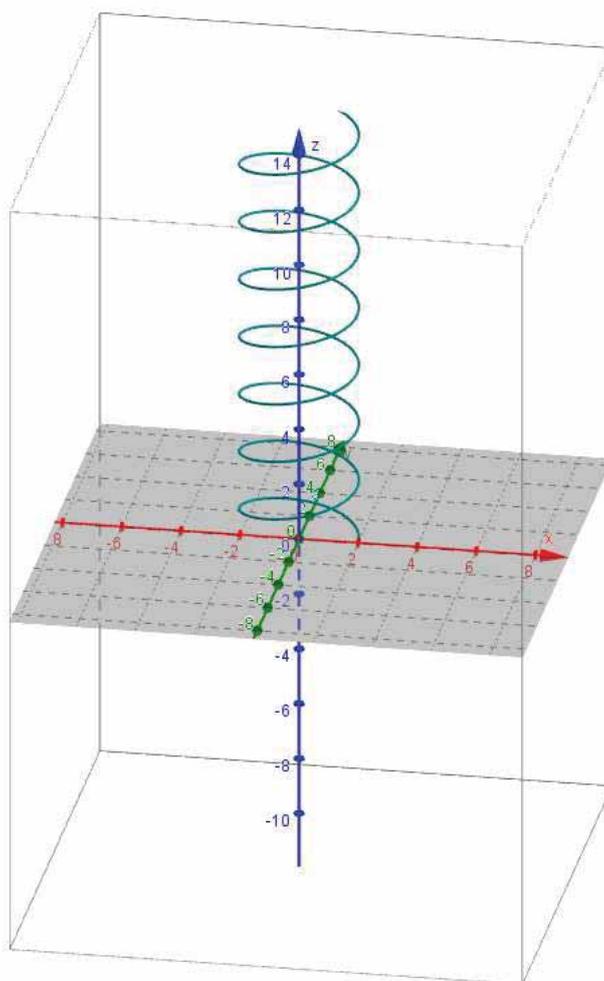


Figura 61 - Representação da equação paramétrica de uma hélice. Fonte: Laboratório Simulação Molecular.

As equações paramétricas são muito utilizadas em varias áreas do conhecimento, como na Física, onde a trajetória de uma partícula é representada por uma equação cujo parâmetro é o tempo.

## REFERÊNCIAS

ADAMS, Colin C. The knot book. An elementary introduction to the mathematical theory of knots. **W. H. Freeman and Company**, New York, 306 p., 1994. Disponível em: <<http://www.ams.org/mathscinet-getitem?mr=1266837>>. Acesso em: 04-11-2013.

ALEXANDER, W. J. Topological invariants of knots and links. **Trans. Amer. Math. Soc.**, Princeton University, Princeton, N. J., v. 30, n. 2, p. 275-306, 1928. Disponível em: <<http://www.ams.org/journals/tran/1928-030-02/S0002-9947-1928-1501429-1/S0002-9947-1928-1501429-1.pdf>>. Acesso em: 04-11-2013.

ANDRAE, D. Molecular knots, links, and fabrics: prediction of existence and suggestion of a synthetic route. *New Journal of Chemistry*, New York, v. 30, n. 6, p. 873-882, 2006. Disponível em: <<http://pub.uni-bielefeld.de/publication/1598770>>. Acesso em: 04-11-2013.

ANFINSEN, C. B. **The molecular basis of evolution**. New York: Wiley, 1959. 228 p.

AUDOLY, B.; CLAUVELIN, N.; NEUKIRCH, S. Elastic knots. **Phys Rev Lett**, New York, v. 99, n. 16, p. 164-301, oct. 2007. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/17995255>>. Acesso em: 04-11-2013.

AR-NATAN, B. *The Knot Atlas*. Recuperado (2013). Disponível em: <<http://katlas.org>>. Acesso em: 04-11-2013.

BERG, J. M.; STRYER, L.; TYMOCZKO, J. L. **Bioquímica**. 6. ed. Rio de Janeiro: Editora Guanabara Koogan Editora Guanabara Koogan, 2008. 672p.

BENACH, J.; SHEN, J.; ROST, B.; XIAO, R.; ACTON, T.; MONTELIONE, G.; HUNT, J. F. **Structure of YBEA from E. coli**. New York, 2011. 300p.

BIRMAN, S. J. New points of view in knot theory. **Bull. Amer. Math. Soc. (N.S.)**, New York, v. 28, n. 2, p. 253-287, 1993. Disponível em: <<http://www.ams.org/journals/bull/2008-45-02/S0273-0979-08-01196-8/>>. Acesso em: 04-11-2013.

BORNSCHLOGL, T. et al. Tightening the knot in phytochrome by single-molecule atomic force microscopy. **Biophys J**, New York, v. 96, n. 4, p. 1508-

1514, 18 feb. 2009. Disponível em: <<http://dx.doi.org/10.1016/j.bpj.2008.11.012>>. Acesso em: 04-11-2013.

BRASHER, R.; SCHAREIN, R. G.; VAZQUEZ, M. New biologically motivated knot table. **Biochem Soc Trans**, New York, v. 41, n. 2, p. 606-611, apr. 2013. Disponível em: <<http://dx.doi.org/10.1042/bst20120278>>. Acesso em: 04-11-2013.

BÖLINGER, D. et al. A Stevedore's Protein Knot. **PLOS Computational Biology**, New York v. 6, n. 4, 2010. Disponível em: <<http://dx.plos.org/10.1371/journal.pcbi.1000731.pdf>>. Acesso em: 04-11-2013.

CALVO, J. A. et al. **Physical knots: knotting, linking, and folding geometric objects in R3: AMS Special Session on Physical Knotting and Unknotting**. Providence: American Mathematical Society, 2002. 340 p.

CAMPBELL, M. K. **Bioquímica Básica**. 3. ed. Porto Alegre, 2001. 300p.

CHAMPE, P. C.; HARVEY, R. A.; FERRIER, D. R. **Bioquímica Ilustrada**. 4. ed. Porto Alegre, 2008. 300p.

Code::Blocks. 2013. Disponível em: < <http://www.codeblocks.org/> >.

DASBACH, T. O.; LIN, X.-S. On the head and the tail of the colored Jones polynomial. *Compositio Math.* [online], v. 142, n. 5, p. 1332–1342, 25 sep. 2006. Disponível em: <<http://journals.cambridge.org/action/displayAbstract?fromPage=online&aid=501884>>. Acesso em: 04-11-2013

DAS, C. et al. **Structural basis for conformational plasticity of the Parkinson's disease-associated ubiquitin hydrolase UCH-L1**. New York, 2011.

DIAMOND, R. **Real-space refinement of the structure of hen egg-white lysozyme**. New York, 2011.

DAVID, L. N.; COX, M. M. Lehninger, Principles of Biochemistry. 5 ed. New York, 2013. Disponível em: <<http://bcs.whfreeman.com/lehninger5e/default.asp?s=&n=&i=&v=&o=&ns=0&id=0&rau=0>>. Acesso em: 04-11-2013.

DAY, I. N.; THOMPSON, R. J. UCHL1 (PGP 9.5): neuronal biomarker and ubiquitin system protein. **Prog Neurobiol**, New York, v. 90, n. 3, p. 327-362, mar. 2010. Disponível em: <<http://dx.doi.org/10.1016/j.pneurobio.2009.10.020>>. Acesso em: [04-11-2013](#).

DZUBIELLA, J. Sequence-specific size, structure, and stability of tight protein knots. **Biophys J**, New York, v. 96, n. 3, p. 831-839, feb. 2009. Disponível em: <<http://dx.doi.org/10.1016/j.bpj.2008.10.019>>. Acesso em: [04-11-2013](#).

ELIAHOU, S.; KAUFFMAN, H. L.; THISTLETHWAITE, B. M. Infinite families of links with trivial Jones polynomial. **Topology** [online], v. 42, n. 1, p. 155-169, jan. 2003. Disponível em: <[http://dx.doi.org/10.1016/S0040-9383\(02\)00012-5](http://dx.doi.org/10.1016/S0040-9383(02)00012-5)>. Acesso em: [04-11-2013](#)

FORGAN, R. S.; SAUVAGE, J. P.; STODDART, J. F. Chemical topology: complex molecular knots, links, and entanglements. **Chem Rev**, New York, v. 111, n. 9, p. 5434-5364, 14 sep. 2011. Disponível em: <<http://dx.doi.org/10.1021/cr200034u>>. Acesso em: [04-11-2013](#).

GRACY, J.; CHICHE, L. Structure and modeling of knottins, a promising molecular scaffold for drug discovery. **Curr Pharm Des**, New York, v. 17, n. 38, p. 4337-4350, dec. 2011. Disponível em: <<http://dx.doi.org/>>. Acesso em: [04-11-2013](#).

HONIG, B. Protein folding: from the Levinthal paradox to structure prediction. **J Mol Biol**, New York, v. 293, n. 2, p. 283-293, 22 oct. 1999. Disponível em: <<http://dx.doi.org/10.1006/jmbi.1999.3006>>. Acesso em: [04-11-2013](#).

HORTON, H. E. A. **Principles of Biochemistry**. New York, 1992. 300p.

HSIEH, S. J. et al. Backbone NMR assignments of a topologically knotted protein in urea-denatured state. **Biomol NMR Assign**, New York, 14 jul. 2013. Disponível em: <<http://dx.doi.org/10.1007/s12104-013-9510-6>>. Acesso em: [04-11-2013](#).

IQF479 - Físico - Química de Proteínas. 2013. Disponível em: <<https://www.siga.ufrj.br/sira/repositorio-curriculo/disciplinas/A6CFAE61-92A4-F713-019B-2364E6992ECF.html>>. Acesso em: [04-11-2013](#).

JIANG, N. et al. Distance-Independent Charge Recombination Kinetics in Cytochrome c-Cytochrome c Peroxidase Complexes: Compensating Changes in the Electronic Coupling and Reorganization Energies. **J Phys Chem B**, New

York, v. 117, n. 31, p. 9129-9141, aug. 2013. Disponível em:  
<<http://www.ncbi.nlm.nih.gov/pubmed/23895339>>. Acesso em: [04-11-2013](#).

JOURNAL MATHEMATICAL MODELLING AND ANALYSIS. [Online].  
**Purchase issue Publication Frequency 5 issues per year.** p. 1648-3510.  
Disponível em: <<http://inga.vgtu.lt/~art/>>. Acesso em: [04-11-2013](#).

JOURNAL OF MATHEMATICAL MODELLING AND APPLICATION. [Online].  
2013. Disponível em: <<http://proxy.furb.br/ojs/index.php/modelling>>. Acesso  
em: [04-11-2013](#).

KANG, T. S.; KINI, R. M. Structural determinants of protein folding. **Cell Mol  
Life Sci**, New York, v. 66, n. 14, p. 2341-2361, jul. 2009. Disponível em:  
<<http://dx.doi.org/10.1007/s00018-009-0023-5>>. Acesso em: [04-11-2013](#).

KATRITCH, V. et al. Properties of ideal composite knots. **Nature**, New York, v.  
388, n. 6638, p. 148-151, 10 jul. 1997. Disponível em:  
<<http://dx.doi.org/10.1038/40582>>. Acesso em: [04-11-2013](#).

KAUFFMAN, H. L. An invariant of regular isotopy. **Trans. Amer. Math. Soc.**  
[online], v. 318, p. 417-471, 1990. Disponível em:  
<<http://www.ams.org/journals/tran/1990-318-02/S0002-9947-1990-0958895-7/>>.  
Acesso em: [04-11-2013](#).

KERNIGHAN, B.; RITCHIE, D. **C: A Linguagem de programação ANSI**. New  
York: Prentice Hall, 1988.

KING, N. P. et al. Structure and folding of a designed knotted protein. **Proc Natl  
Acad Sci U S A**, New York, v. 107, n. 48, p. 20732-20737, 30 nov. 2010.  
Disponível em: <<http://dx.doi.org/10.1073/pnas.1007602107>>. Acesso em: [04-11-2013](#).

KOHL, N. **cppreference.com**. 2013. Disponível em:  
<<http://en.cppreference.com/w/>>. Acesso em: [04-11-2013](#).

KOLESOV, G. et al. Protein knot server: detection of knots in protein structures.  
**Nucleic Acids Res, Oxford Journals** [online], v. 35, n. Web Server issue, p.  
W425-428, jul. 2007. Disponível em: <<http://dx.doi.org/10.1093/nar/gkm312>>.  
Acesso em: [04-11-2013](#).

KONIARIS, K.; MUTHUKUMAR, M. Knottedness in ring polymers. **Physical  
Review Letters**, New York, v. 66, p. 2211-2214, apr. 1991. Disponível em:

<<http://adsabs.harvard.edu/abs/1991PhRvL..66.2211K>>. Acesso em: [04-11-2013](#).

KRAFFT, C. F. **Can science explain life?** Lancaster: Science Press Printing Company, 1931. 98 p.

LAI, Y. L.; CHEN, C. C.; HWANG, J. K. pKNOT v.2: the protein KNOT web server. **Nucleic Acids Res, Oxford Journals** [online], v. 40, n. Web Server issue, p. W228-231, jul. 2012. Disponível em: <<http://dx.doi.org/10.1093/nar/gks592>>. Acesso em: [04-11-2013](#).

LEHNINGER, A. L. **Princípios de bioquímica**. New York, 1985. 300p.

LUA, R. C.; GROSBURG, A. Y. Statistics of knots, geometry of conformations, and evolution of proteins. **PLoS Comput Biol** [online], v. 2, n. 5, p. e45, may 2006. Disponível em: <<http://dx.doi.org/10.1371/journal.pcbi.0020045>>. Acesso em: [04-11-2013](#).

MALLAM, A. L. How does a knotted protein fold? **Febs j** [online], v. 276, n. 2, p. 365-375, jan. 2009. Disponível em: <<http://dx.doi.org/10.1111/j.1742-4658.2008.06801.x>>. Acesso em: [04-11-2013](#).

MALLAM, A. L.; JACKSON, S. E. Folding studies on a knotted protein. **J Mol Biol** [online], v. 346, n. 5, p. 1409-1421, 11 mar. 2005. Disponível em: <<http://dx.doi.org/10.1016/j.jmb.2004.12.055>>. Acesso em: [04-11-2013](#).

MALLAM, A. L. et al. Knotted fusion proteins reveal unexpected possibilities in protein folding. **Mol Cell** [online], v. 30, n. 5, p. 642-648, 6 jun. 2008. Disponível em: <<http://dx.doi.org/10.1016/j.molcel.2008.03.019>>. Acesso em: [04-11-2013](#).

MARZOCCO, A.; TORRES, B. B. **Bioquímica Básica**. 3. ed. Rio de Janeiro: 2007.

MICHELSON, A. A.; MORLEY, E. W. On the Relative Motion of the Earth and the Luminiferous Ether. **American Journal of Science**, New York, volume 03, edição, p. 449-463, 2007.

MELUZZI, D.; SMITH, D. E.; ARYA, G. Biophysics of knotting. **Annu Rev Biophys**, New York, v. 39, p. 349-66, 2010. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/20192771>>. Acesso em: [04-11-2013](#).

MICROSOFT. Microsoft Office - Office.com. 2013. Disponível em: <<http://office.microsoft.com/pt-br/>>. Acesso em: [04-11-2013](#).

MILLETT, K. C. et al. Identifying knots in proteins. **Biochem Soc Trans**, New York, v. 41, n. 2, p. 533-7, apr. 2013. Disponível em: <<http://dx.doi.org/10.1042/bst20120339>>. Acesso em: [04-11-2013](#).

MOGLICH, A.; JODER, K.; KIEFHABER, T. End-to-end distance distributions and intrachain diffusion constants in unfolded polypeptide chains indicate intramolecular hydrogen bond formation. **Proc Natl Acad Sci U S A** [online], v. 103, n. 33, p. 12394-12399, 15 aug. 2006. Disponível em: <<http://dx.doi.org/10.1073/pnas.0604748103>>. Acesso em: [04-11-2013](#).

ORIGINLAB. OriginLab - Origin and OriginPro - Data Analysis and Graphing Software. 2013. Disponível em: <<http://www.originlab.com/>>. Acesso em: [04-11-2013](#).

OUELLETTE, B. F. F.; BAXEVANIS, A. D. **Bioinformatics**: a practical guide to the analysis of genes and proteins. New York: John Wiley, 1998. 370 p.

OXFORD JOURNALS. Life Sciences. Nucleic Acids Research. **Submitting to the Web Server Issue**. 2013. Disponível em: <[http://www.oxfordjournals.org/our\\_journals/nar/for\\_authors/submission\\_webserver.html](http://www.oxfordjournals.org/our_journals/nar/for_authors/submission_webserver.html)>. Acesso em: [04-11-2013](#).

PDB. RCSB Protein Data Bank - RCSB PDB. 2013. Disponível em: <[http://www.rcsb.org/pdb/static.do?p=general\\_information/pdb\\_statistics/index.html](http://www.rcsb.org/pdb/static.do?p=general_information/pdb_statistics/index.html)>. Acesso em: [04-11-2013](#).

QIAN, B. et al. High-resolution structure prediction and the crystallographic phase problem. **Nature** [online], v. 450, n. 7167, p. 259-264, 14 oct. 2007. Disponível em: <<http://www.nature.com/nature/journal/v450/n7167/abs/nature06249.html>>. Acesso em: [04-11-2013](#).

RasMol and OpenRasMol. 2013. Disponível em: <<http://rasmol.org/>>. Acesso em: [04-11-2013](#).

- RAWDON, E. J. et al. Knot localization in proteins. **Biochem Soc Trans** [online], v. 41, n. 2, p. 538-41, apr. 2013. Disponível em: <<http://dx.doi.org/10.1042/bst20120329>>. Acesso em: [04-11-2013](#).
- ROYCHAUDHURI, R. et al. C-terminal turn stability determines assembly differences between A $\beta$ 40 and A $\beta$ 42. **J Mol Biol** [online], v. 425, n. 2, p. 292-308, jan. 2013. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/23154165>>. Acesso em: [04-11-2013](#).
- RUVINSKY, A. M. et al. Structure fluctuations and conformational changes in protein binding. **J Bioinform Comput Biol** [online], v. 10, n. 2, p. 124, apr. 2012. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/22809338>>. Acesso em: [04-11-2013](#).
- SACKHEIM, G. I.; LEHMAN, D. D. **Química e Bioquímica para ciências biomédicas**. 8. ed. São Paulo: 2001. 300p
- SCHMIDBERGER, J.W. et al. **Structure of a Group I haloacid dehalogenase from Pseudomonas putida strain PP3**. 2011.
- SHASTRI, A. Polynomial Representation of Knots. **Tôhoku Math J.**, New York, v. 44, edição 2, p. 11-17, 1992.
- SKRBIĆ, T.; MICHELETTI, C.; FACCIOLI, P. The Role of Non-Native Interactions in the Folding of Knotted Proteins. **PLOS Computational Biology**, New York, v. 8, n. 6, 14 jun. 2012. Disponível em: <<http://dx.plos.org/10.1371/journal.pcbi.1002504.pdf>>. Acesso em: [04-11-2013](#)
- SOUZA, M. et al. Solving the molecular distance geometry problem with inaccurate distance data. **BMC Bioinformatics** [online], v. 14, Suppl 9, p. S7, jun. 2013. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/23901894>>. Acesso em: [04-11-2013](#).
- SULKOWSKA, I. J. et al. Knotting pathways in proteins. **Biochem Soc Trans** [online], v. 41, n. 2, p. 523-527, apr. 2013. Disponível em: <<http://dx.doi.org/10.1042/bst20120342>>. Acesso em: [04-11-2013](#).
- \_\_\_\_\_. Conservation of complex knotting and slipknotting patterns in proteins. **Proc Natl Acad Sci U S A** [online], v. 109, n. 26, p. E1715-23, 26 jun. 2012. Disponível em: <<http://dx.doi.org/10.1073/pnas.1205918109>>. Acesso em: [04-11-2013](#).

SUMMIT, S. **C Programming**. 2013. Disponível em: <<http://www.eskimo.com/~scs/cclass/>>. Acesso em: [04-11-2013](#).

SUŁKOWSKA, I. J.; SUŁKOWSKI, P.; ONUCHIC, J. **Dodging the crisis of folding proteins with knots**. 3 mar. 2009. Disponível em: <<http://www.pnas.org/content/106/9/3119>>. Acesso em: [04-11-2013](#).

SUŁKOWSKA, I. J. et al. **Stabilizing effect of knots on proteins**. 16 dec. 2008. Disponível em: <<http://www.pnas.org/content/105/50/19714.full>>. Acesso em: [04-11-2013](#).

SZYMCZAK, P. Tight knots in proteins: can they block the mitochondrial pores? **Biochem Soc Trans** [online], v. 41, n. 2, p. 620-624, apr. 2013. Disponível em: <<http://dx.doi.org/10.1042/bst20120261>>. Acesso em: [04-11-2013](#).

Kumarevel, T.S., Karthe, P., Kuramitsu, S., Yokoyama, S, **Crystal Structure of a Hypothetical Protein(MJ0366) from Methanocaldococcus jannaschii**, New York, **2011**. 300p.

Thomson, W. On Vortex Motion. **Transactions of the Royal Society of Edinburgh**, New York, v. 25, edição 2, 1868, p.217 – 260.

THE KNOTPLOT SITE. [online]. 2013. Disponível em: <<http://www.knotplot.com/>>. Acesso em [04-11-2013](#).

Wagner, J.R., Brunzelle, J.S., Forest, K.T., Vierstra, R.D. **A light-sensing knot revealed by the structure of the chromophore-binding domain of phytochrome**, New York, **2011**) 300p.

VIRNAU, P. **Knots in the proteins - prediction server**. [online]. 2013. Disponível em: < <http://knots.mit.edu/> >. Acesso em [04-11-2013](#).

VIRNAU, P.; MALLAM, A.; JACKSON, S. Structures and folding pathways of topologically knotted proteins. **J Phys Condens Matter**, v. 23, n. 3, p. 033101, Jan 26 2011. Disponível em: <<http://dx.doi.org/10.1088/0953-8984/23/3/033101>>. Acesso em [04-11-2013](#).

VIRNAU, P.; MIRNY, A. L.; KARDAR, M. Intricate Knots in Proteins: Function and Evolution. **PLOS Computational Biology**, v. 2, n. 9, 2006/9/15 2006. Disponível em: < <http://dx.plos.org/10.1371/journal.pcbi.0020122.pdf> > Acesso em [04-11-2013](#).

VOET, D.; VOET, J. G. **Bioquímica**. 3. ed. Porto Alegre. 2006.

ZHANG, J. et al. Analysis of the relationship between end-to-end distance and activity of single-chain antibody against colorectal carcinoma. **Theoretical Biology and Medical Modelling**, v. 9, n. 1, p. 38, 2012-08-22 2012. Disponível em: < <http://www.tbiomed.com/content/9/1/38> >. Acesso em [04-11-2013](#).

## 7.0 Anexos

Programação de automação de cálculos de distâncias internas e cálculo dos diedros

```
#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

#include <math.h>

#include "funcoes.h"

int main()

{

    FILE *origem,*atomos_CA,*coordenadas,*baseCoordenadas,*Distancias, *destino,

    *arqAngulo, *teste;

    char dados[50];

    char linha[81];

    char auxLinha[81];

    char auxlinhaCA[81];

    char string[100];

    int primeiroAtomo=1;

    tcoordenadas regUm, regDois;

    int contador = 0;

    while (TRUE)

    {

        system("CLS");

        printf("Digite o nome do arquivo de origem: ");
```

```
scanf("%s", dados);

if ((origem = fopen(dados, "r")) != NULL)
{
    break;
}

printf("ARQUIVO NAO ENCONTRADO\nTECLE ALGO...");

getch();
}

atomos_CA = fopen("atomos_CA.txt", "w");
coordenadas = fopen("coordenadas.txt", "w");
baseCoordenadas = fopen("base.txt", "w");
Distância s = fopen("Distância s.txt", "w");

// Inicia estrutura auxiliar.
regDois.atomo = 0;
regDois.x = 0.0;
regDois.y = 0.0;
regDois.z = 0.0;
regDois.Distância = 0.0;

while (fgets(linha, 81, origem)!=NULL)
{
    strcpy(auxLinha,linha);
    strcpy(auxlinhaCA,linha);

    if (strcmp(substring(auxLinha,0,4), ATOM) == 0 &&
        strcmp(substring(auxlinhaCA,13,2),CA)== 0 ) // se a linha for iniciada por ATOM...
```

```
{  
    // Coleta os dados necessários do arquivo de origem.  
    strcpy(auxLinha,linha);  
    regUm.atomo = atoi(substring(auxLinha, 7, 6));  
    strcpy(auxLinha,linha);  
    regUm.x = atof(substring(auxLinha, 31, 7));  
    strcpy(auxLinha,linha);  
    regUm.y = atof(substring(auxLinha, 40, 7));  
    strcpy(auxLinha,linha);  
    regUm.z = atof(substring(auxLinha, 47, 7));  
  
    // Calcula a distância  
    regUm.Distância = calcDistância (regUm.x, regUm.y, regUm.z, regDois.x, regDois.y,  
regDois.z);  
  
    if (primeiroAtomo == 1) //Se for o primeiro átomo  
    {  
        primeiroAtomo=0;  
        regUm.Distância = 0.0;  
    }  
  
    regDois = regUm;  
  
    // Escreve nos arquivos de destino.  
    sprintf(string,"%s\n", linha);  
    //trocaFlutuante(string);  
    fprintf(atomos_CA, "%s",string);
```

```
    sprintf(string, "%s %5d %10.3f %10.3f %10.3f\n", ATOM, regUm.atomo, regUm.x,
regUm.y, regUm.z);

    //trocaFlutuante(string);

    fprintf(coordenadas, "%s", string);

    sprintf(string, "%s %5d %10.3f %10.3f %10.3f\n", ATOM, regUm.atomo, regUm.x,
regUm.y, regUm.z);

    //trocaFlutuante(string);

    fprintf(baseCoordenadas, "%s", string);

    sprintf(string, "%5d %10.3f\n", regUm.atomo, regUm.Distância );

    //trocaFlutuante(string);

    fprintf(Distância s, "%s", string);

}

}

fclose(origem);

fclose(atomos_CA);

fclose(coordenadas);

fclose(baseCoordenadas);

fclose(Distância s);

// Mensagem de Conclusão da fase 1

system("CLS");

printf("*****\n");

printf("*** OS ARQUIVOS BASE FORAM CRIADOS COM SUCESSO ***\n");

printf("*****\n\n");

printf("Tecle algo para continuar...");
```

```
getch();

system("CLS");

/*=====*/
/* Início da rotina de criação dos arquivos de pesquisa */
/* baseado em informações do usuário */
/*=====*/

// Variáveis necessárias à Fase 2.

int CaReferencia = 0; // Recebe o número do átomo a ser usado como
//referência para a criação dos novos arquivos

unsigned int HaReferencia = 0; // Controle que verifica se o número
// do átomo digitado, existe no arquivo

char nomeDestino[] = "dist_"; // Prefixo do nome dos arquivos
// que serão criados

char extensaoDestino[] = ".csv"; // Extensão dos arquivos
// que serão criados

char opt, exhibe;

exibe = '*';

tAngulo angulo[4];

int anguloLinha = 1;

float Ux, Uy, Uz;

float Vx, Vy, Vz;

float prodVet; // Guarda Resultado da Segunda Etapa

float raizU, raizV, prodUV; // Guarda Resultado da Terceira Etapa

float anguloFinal;
```

```
// Menu
while (TRUE)
{
    while (TRUE)
    {
        system("CLS");

        printf("*****\n");
        printf("*** ### *** MENU *** ### ***\n");
        printf("*****\n");
        printf("1 -> Distância s Relativas - Gerar Automaticamente\n");
        printf("2 -> Distância s Relativas - Gerar Manualmente\n");
        printf("3 -> Gerar Angulos\n");
        printf("4 -> Gerar Angulos por Crammer\n");
        printf("0 -> Sair\n");
        printf("=====\n");
        printf("Digite a Opcao (1, 2, 3, 4, 0) -> ");

        // scanf("%d",&opt);

        opt = getch();

        if (opt == '1' || opt == '2' || opt == '3' || opt == '4' || opt == '0')
        {
            break;
        }
    }
}

if (opt == '1') // Geração automática dos arquivos
```

```
{  
    // Mensagem de Espera  
    system("CLS");  
    printf("Aguarde a Criação dos arquivos... ");  
  
    // Cria e abre arquivo de Teste  
    teste = fopen("teste.txt", "w");  
  
    // Abre o Arquivo base.txt para leitura "r"  
    baseCoordenadas = fopen("base.txt", "r");  
  
    while (fgets(linha, 45, baseCoordenadas) != NULL)  
    {  
        printf("%c", exhibe);  
        contador++;  
        // Captura número do átomo base atual  
        strcpy(auxLinha, linha);  
        regUm.atomo = atoi(substring(auxLinha,5,6));  
  
        // Captura coordenadas do átomo base atual  
        strcpy(auxLinha,linha);  
        regUm.x = atof(substring(auxLinha, 14, 7));  
        strcpy(auxLinha,linha);  
        regUm.y = atof(substring(auxLinha, 25, 7));  
        strcpy(auxLinha,linha);  
        regUm.z = atof(substring(auxLinha, 36, 7));  
  
        // *****
```

```
fprintf(teste, "%d -> ATOMO = %d <-> X = %f <-> Y = %f <-> Z = %f\n",
        contador, regUm.atomo, regUm.x, regUm.y, regUm.z);

// *****

// Cria nome para o novo Arquivo de Distância Relativa
strcpy(auxLinha,linha);

strcpy(dados, strcat(nomeDestino, trim(substring(auxLinha, 5, 6))));

strcat(dados,extensaoDestino);

// Abre o arquivo coordenadas para leitura "r"
coordenadas = fopen("coordenadas.txt", "r");

// Abre o arquivo de destino para escrita "w"
destino = fopen(dados, "w");

// Inicia looping secundário (arquivo coordenadas.txt)
// void lerArquivoDistância sAbsolutas();
while(fgets(linha, 45, coordenadas) != NULL)
{

    // captura as coordenadas
    // do registro corrente
    strcpy(auxLinha,linha);

    regDois.atomo = atoi(substring(auxLinha, 5, 6));

    strcpy(auxLinha,linha);

    regDois.x = atof(substring(auxLinha, 14, 7));

    strcpy(auxLinha,linha);
```

```
    regDois.y = atof(substring(auxLinha, 25, 7));

    strcpy(auxLinha,linha);

    regDois.z = atof(substring(auxLinha, 36, 7));

    // Faz o Calculo da Distância

    // entre o átomo de referência

    // e os outros e Grava em um novo Arquivo

    regDois.Distância = calcDistância (regUm.x, regUm.y, regUm.z, regDois.x, regDois.y,
regDois.z);

    // Grava as informações no arquivo de destino

    sprintf(string, "%5d;%10.3f\n", regDois.atomo, regDois.Distância );

    trocaFlutuante(string);

    fprintf(destino, "%s",string);
}

// fecha os arquivos

fclose(coordenadas);

fclose(destino);

strcpy(nomeDestino,"dist_");

}

// fecha último arquivo

fclose(baseCoordenadas);

// Mensagem de término do procsso.

system("CLS");

printf("Arquivos criados com Sucesso.\nPor Favor Tecle Algo Para Retornar ao
Menu...");
```

```
    getch();
}
else if (opt == '2') // Geração manual de arquivos
{
    do
    {
        // Entrada
        system("CLS");

        printf("*****\n");
        printf("Rastreando Distância s\n");
        printf("*****\n\n");
        printf("Para sair digite o valor zero\n\n");
        printf("CA Referencia: ");
        scanf("%d", &CaReferencia);

        // Abertura do arquivo base
        coordenadas = fopen("coordenadas.txt", "r");

        // Verifica a Existência dos CAs digitados
        // fazendo a verificação do arquivo Distância s.txt
        while(fgets(linha, 45, coordenadas) != NULL)
        {
            strcpy(auxLinha, linha);

            regUm.atomo = atoi(substring(auxLinha, 5, 6));

            if (CaReferencia == regUm.atomo) // átomo de referência encontrado
            {
                HaReferencia = 1; // CA Referencia encontrado
            }
        }
    }
}
```

```
// Coleta coordenadas de referência
strcpy(auxLinha,linha);
regUm.x = atof(substring(auxLinha, 14, 7));
strcpy(auxLinha,linha);
regUm.y = atof(substring(auxLinha, 25, 7));
strcpy(auxLinha,linha);
regUm.z = atof(substring(auxLinha, 36, 7));
// *****
printf("X = %f - Y = %f - Z = %f\n\n",
      regUm.x, regUm.y, regUm.z);
// *****

// Sai do looping
break;
}
else
{
    HaReferencia = 0;
}
}

fclose(coordenadas); // fecha o arquivo coordenadas.txt

if (HaReferencia == 1) // Tudo em ordem
{
    // Zera o flag para a próxima pesquisa
    HaReferencia = 0;
}
```

```
// Cria nome para o novo Arquivo de Distância Relativa
strcpy(auxLinha,linha);
strcpy(dados, strcat(nomeDestino, trim(substring(auxLinha, 5, 6))));
strcat(dados,extensaoDestino);

// Cria e abre o novo
// arquivo de Distância Relativa
// no modo de Escrita
destino = fopen(dados, "w");

// Escreve o cabeçalho do arquivo
sprintf(string, "Distância s Relativas entre o atomo %5d e os outros\n\n",
regUm.atomo);
trocaFlutuante(string);
fprintf(destino, "%s", string);

// Abertura do arquivo base
coordenadas = fopen("coordenadas.txt", "r");

// void lerArquivoDistância sAbsolutas();

// Inicia a leitura do Arquivo
// de Distâncias Absolutas
while(fgets(linha, 45, coordenadas) != NULL)
{
    // captura as coordenadas
```

```
// do registro corrente
strcpy(auxLinha,linha);
regDois.atomo = atoi(substring(auxLinha, 5, 6));
strcpy(auxLinha,linha);
regDois.x = atof(substring(auxLinha, 14, 7));
strcpy(auxLinha,linha);
regDois.y = atof(substring(auxLinha, 25, 7));
strcpy(auxLinha,linha);
regDois.z = atof(substring(auxLinha, 36, 7));

// Faz o Calculo da Distância
// entre o átomo de referência
// e os outros e Grava em um novo Arquivo
regDois.Distância = calcDistância (regUm.x, regUm.y, regUm.z, regDois.x,
regDois.y, regDois.z);

// Grava as informações no arquivo de destino
sprintf(string, "%5d;%10.3f\n", regDois.atomo, regDois.Distância );
trocaFlutuante(string);
fprintf(destino, "%s",string);
}

// fecha os arquivos
fclose(coordenadas);
fclose(destino);
strcpy(nomeDestino,"dist_");
```

```
// Exibe mensagem de término do processo

printf("Arquivo criado com sucesso!!!");

printf("Tecla algo para continuar!!!\n\n");

getch();

}

else // Problemas

{

    if(CaReferencia != 0)

    {

        printf("Atomo Final Invalido!!!!\n");

        printf("Tecla algo para continuar!!!\n\n");

        getch();

    }

}

while (CaReferencia != 0);

}

else if (opt == '3')

{

    arqAngulo = fopen("angulo.csv", "w");

    coordenadas = fopen("coordenadas.txt", "r");

    contador = 0;

    while (fgets(linha, 45, coordenadas) != NULL)

    {

        // captura as coordenadas

        // do registro corrente
```

```
strcpy(auxLinha,linha);

angulo[contador].x = atof(substring(auxLinha, 14, 7));

strcpy(auxLinha,linha);

angulo[contador].y = atof(substring(auxLinha, 25, 7));

strcpy(auxLinha,linha);

angulo[contador].z = atof(substring(auxLinha, 36, 7));

// *****

//printf("Contador: %d - Linha Ang: %d\n", contador, anguloLinha);

//system("PAUSE");

// *****

contador++;

if (contador == 4) // Calcula angulo
{
    // zera contador

    contador = 0;

    // Executa fórmula

    // Primeira Etapa

    Ux = angulo[1].x - angulo[0].x;

    Uy = angulo[1].y - angulo[0].y;

    Uz = angulo[1].z - angulo[0].z;

    //printf("\nUx = %f - Uy = %f - Uz = %f\n", Ux, Uy, Uz);

    Vx = angulo[3].x - angulo[2].x;
```

```
Vy = angulo[3].y - angulo[2].y;

Vz = angulo[3].z - angulo[2].z;

//printf("Vx = %f - Vy = %f - Vz = %f\n", Vx, Vy, Vz);

// Segunda Etapa

prodVet = (Ux * Vx) + (Uy * Vy) + (Uz * Vz);

//printf("Produto Vetorial = %f\n", prodVet);

// Terceira Etapa

raizU = sqrt(pow(Ux, 2) + pow(Uy, 2) + pow(Uz, 2));

raizV = sqrt(pow(Vx, 2) + pow(Vy, 2) + pow(Vz, 2));

prodUV = raizU * raizV;

//printf("Produto UV = %f\n", prodUV);

// Quarta Etapa

anguloFinal = prodVet / prodUV;

//printf("Antes do cosseno = %f\n", anguloFinal);

anguloFinal = acos(anguloFinal)*180/PI;

//printf("Angulo Final = %f\n",anguloFinal);

//system("PAUSE");

sprintf(string, "%d;%f\n", anguloLinha, anguloFinal);

trocaFlutuante(string);

fprintf(arqAngulo, "%s", string);

anguloLinha++;

}

}

fclose(arqAngulo);
```

```
fclose(coordenadas);

system("CLS");

if (contador == 0)
{
    printf("O Conteudo do Arquivo eh Multiplo de 4\n");
}

else
{
    printf("Restaram %d Atomos\n\n", contador);
}

printf("Tecla Algo para Continuar . . .");
system("PAUSE > NULL");
}

else if (opt == '4')
{
    /*******

    printf("\n");

    /*******

    // Define estruturas

    tAngulo structCoord[3];

    tDeterminantes structDet1, structDet2;

    // Abre Arquivo de origem e cria arquivo de destino (Crammer)

    origem = fopen("base.txt", "r");

    destino = fopen("crammer.csv", "w");

    for (contador = 0; contador <= 2; contador++)

    {
```

```
fgets(linha, 45, origem);

// captura as coordenadas
// do registro corrente
strcpy(auxLinha,linha);
structCoord[contador].x = atof(substring(auxLinha, 14, 7));
strcpy(auxLinha,linha);
structCoord[contador].y = atof(substring(auxLinha, 25, 7));
strcpy(auxLinha,linha);
structCoord[contador].z = atof(substring(auxLinha, 36, 7));
}

// Chama a função de cálculo de determinante
structDet1.detPrincipal = determinante(structCoord[0], structCoord[1],
structCoord[2],'n');

structDet1.detA = determinante(structCoord[0], structCoord[1], structCoord[2],'x');
structDet1.detB= determinante(structCoord[0], structCoord[1], structCoord[2],'y');
structDet1.detC = determinante(structCoord[0], structCoord[1], structCoord[2],'z');

while(fgets(linha, 45, origem) != NULL)
{
    structCoord[0] = structCoord[1];
    structCoord[1] = structCoord[2];

    strcpy(auxLinha,linha);
    structCoord[2].x = atof(substring(auxLinha, 14, 7));
    strcpy(auxLinha,linha);
```

```
    structCoord[2].y = atof(substring(auxLinha, 25, 7));

    strcpy(auxLinha,linha);

    structCoord[2].z = atof(substring(auxLinha, 36, 7));

    // Chama a função de cálculo de determinante

    structDet2.detPrincipal = determinante(structCoord[0], structCoord[1],
    structCoord[2], 'n');

    structDet2.detA = determinante(structCoord[0], structCoord[1],
    structCoord[2], 'x');

    structDet2.detB = determinante(structCoord[0], structCoord[1],
    structCoord[2], 'y');

    structDet2.detC = determinante(structCoord[0], structCoord[1],
    structCoord[2], 'z');

    // Chama a função que calcula o ângulo através de Cramer

    anguloFinal = anguloCramer(structDet1, structDet2);

    // Grava o resultado final no arquivo

    sprintf(string, "%d;%f\n", anguloLinha, anguloFinal);

    trocaFlutuante(string);

    fprintf(destino, "%s", string);

    anguloLinha++;

    structDet1 = structDet2;

}

fclose(origem);

fclose(destino);

printf("Tecla Algo para Continuar . . .");

system("PAUSE > NULL");
```

```
}  
else  
{  
    system("CLS");  
    printf("Deseja Sair do Sistema (s/n)?");  
    opt = getch();  
    opt = toupper(opt);  
    if (opt == 'S')  
    {  
        break;  
    }  
}  
}  
  
// Finaliza e sai do sistema  
return 0;  
}
```

Determinando as funções

/ Prototipação de funções.

```
char * substring(char *origem, int inicio, int quant);
```

```
char *ltrim(char *s);
```

```
char *rtrim(char *s);
```

```
char *trim(char *s);
```

```
float calcDistância (float x1, float y1, float z1, float x2, float y2, float z2);
```

```
float determinante(tAngulo a1, tAngulo a2, tAngulo a3, char ch);
```

```
float anguloCrammer(tDeterminantes d1, tDeterminantes d2);
```

```
void trocaFlutuante(char *string);
```

```
// void lerArquivoDistância sAbsolutas(void);
```

Funções para extrair

```
include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

#include "funcoes.h"

// Função que extrai substrings de uma string
char * substring(char *origem, int inicio, int quant)
{
    char *res = origem;
    int i = 0;

    // posição inicial menor que 0 ou
    // posição inicial muito exagerada?
    if((inicio < 0) || (inicio > strlen(origem)))
    {
        inicio = 0;
    }

    // quantidade de caracteres muito exagerada?
    if(quant > inicio + strlen(origem))
    {
        quant = strlen(origem) - inicio;
    }
}
```

```
// obtem os caracteres desejados
for(i = 0; i <= quant - 1; i++)
{
    res[i] = origem[inicio + i];
}

// marca o fim da string
res[i] = '\0';

return res;
}
```

```
// Funções TRIM
```

```
char *ltrim(char *s)
{
    while(isspace(*s)) s++;
    return s;
}
```

```
char *rtrim(char *s)
{
    char* back = s + strlen(s);
    while(isspace(*--back));
    *(back+1) = '\0';
    return s;
}
```

```
}
```

```
char *trim(char *s)
```

```
{
```

```
    return rtrim(ltrim(s));
```

```
}
```

```
// função que calcula as distâncias entre os átomos no espaço (coordenadas x, y, z).
```

```
float calcDistância (float x1, float y1, float z1, float x2, float y2, float z2)
```

```
{
```

```
    x2 = pow((x2 - x1), 2);
```

```
    y2 = pow((y2 - y1), 2);
```

```
    z2 = pow((z2 - z1), 2);
```

```
    return sqrt(x2 + y2 + z2);
```

```
}
```

```
// Troca os pontos por vírgulas
```

```
void trocaFlutuante(char *string)
```

```
{
```

```
    unsigned char ch;
```

```
    int i = 0;
```

```
    while ((ch = string[i]) != '\0')
```

```
    {
```

```
        // printf ("%d - %c\t",ch, ch);
```

```
        if (ch == 46) // ponto
```

```
        {
```

```
            ch = 44; // vírgula
```

```
    }  
    string[i] = ch;  
    i++;  
}  
}
```

```
float determinante(tAngulo a1, tAngulo a2, tAngulo a3, char ch)
```

```
{  
    float det, p1, p2, p3, s1, s2, s3, inv = -1;  
    switch (ch)  
    {  
        case 'x':  
            a1.x = 1;  
            a2.x = 1;  
            a3.x = 1;  
            break;  
        case 'y':  
            a1.y = 1;  
            a2.y = 1;  
            a3.y = 1;  
            break;  
        case 'z':  
            a1.z = 1;  
            a2.z = 1;  
            a3.z = 1;  
            break;  
        default:
```

```
        break;
    }

/*
printf("%f\t%f\t%f\n", a1.x, a1.y, a1.z);
printf("%f\t%f\t%f\n", a2.x, a2.y, a2.z);
printf("%f\t%f\t%f\n", a3.x, a3.y, a3.z);
*/

// Diagonal Principal
p1 = (a1.x * a2.y * a3.z);
p2 = (a1.y * a2.z * a3.x);
p3 = (a1.z * a2.x * a3.y);

// Diagonal Secundária
s1 = inv * (a1.z * a2.y * a3.x);
s2 = inv * (a1.x * a2.z * a3.y);
s3 = inv * (a1.y * a2.x * a3.z);

// Determinante
det = p1 + p2 + p3 + s1 + s2 + s3;

/*
printf("P1 = %f - P2 = %f - P3 = %f - S1 = %f - S2 = %f - S3 = %f - Determinante = %f\n",
p1, p2, p3, s1, s2, s3, det);

system("PAUSE");
*/
```

```
        return det;
    }

float anguloCrammer(tDeterminantes d1, tDeterminantes d2)
{
    float angulo;

    angulo = d1.detA * d2.detA + d1.detB * d2.detB + d1.detC * d2.detC;

    angulo /= ((sqrt(pow(d1.detA,2) + pow(d1.detB,2) + pow(d1.detC,2))) *
               (sqrt(pow(d2.detA,2) + pow(d2.detB,2) + pow(d2.detC,2))));

    angulo = acos(angulo)*180/PI;

    return angulo;
}

// Inicia a leitura do Arquivo
// de Distâncias Absolutas
/* void lerArquivoDistância sAbsolutas()
{
    while(fgets(linha, 45, coordenadas) != NULL)
    {

        // captura as coordenadas
        // do registro corrente
```

```
strcpy(auxLinha,linha);

regDois.atomo = atoi(substring(auxLinha, 5, 6));

strcpy(auxLinha,linha);

regDois.x = atof(substring(auxLinha, 14, 7));

strcpy(auxLinha,linha);

regDois.y = atof(substring(auxLinha, 25, 7));

strcpy(auxLinha,linha);

regDois.z = atof(substring(auxLinha, 36, 7));

// Faz o Calculo da Distância

// entre o átomo de referência

// e os outros e Grava em um novo Arquivo

regDois.Distância = calcDistância (regUm.x, regUm.y, regUm.z, regDois.x, regDois.y,
regDois.z);

// Grava as informações no arquivo de destino

fprintf(destino,"%5d;%10.3f\n", regDois.atomo, regDois.Distância );

}

}

/
```