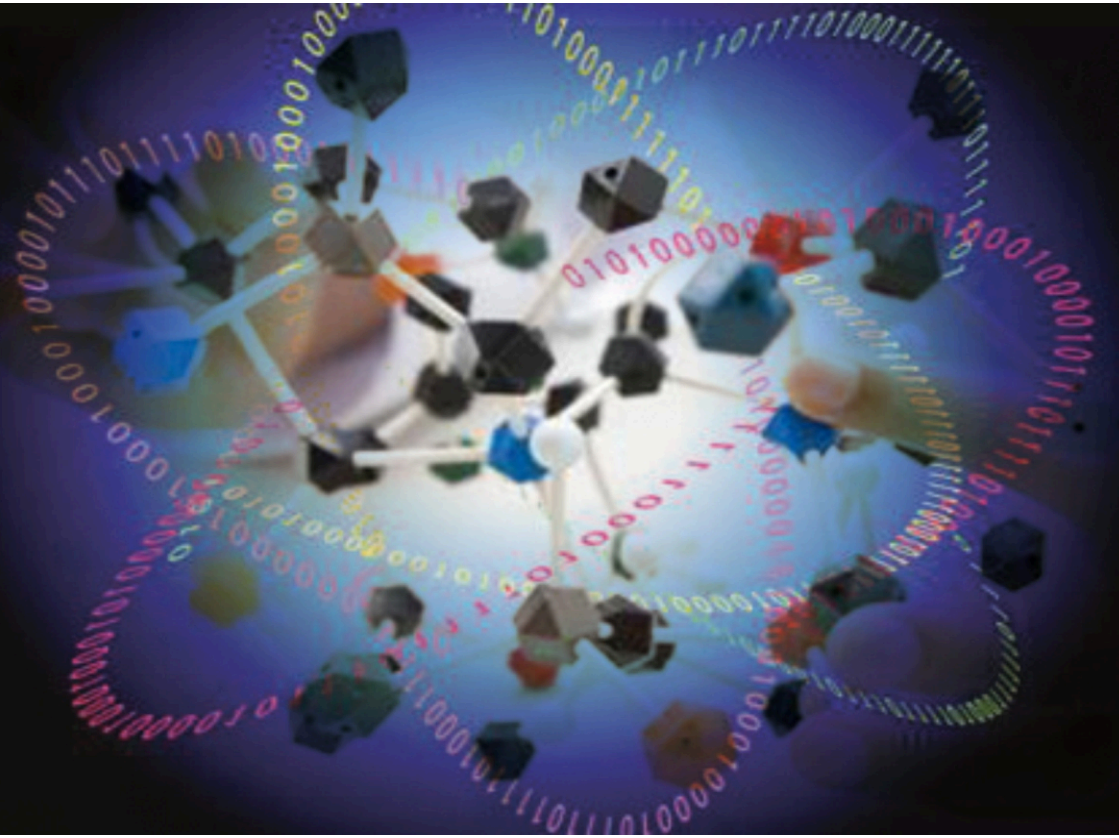
Boletim de Pesquisa 17
e Desenvolvimento Novembro, 2007

ISSN 1677-9266

Cálculo Analítico de Parâmetros Estruturais de Proteínas usando a Teoria Alpha Shapes



Embrapa

ISSN 1677-9266
Novembro, 2007

Empresa Brasileira de Pesquisa Agropecuária
Embrapa Informática Agropecuária
Ministério da Agricultura, Pecuária e Abastecimento

Boletim de Pesquisa e Desenvolvimento 17

Cálculo Analítico de Parâmetros Estruturais de Proteínas usando a Teoria Alpha Shapes

George Barreto Perreira Bezerra
Michel Eduardo Beleza Yamagishi
Fábio Danilo Vieira
Edgard Henrique dos Santos
Marcelo Gonçalves Narciso
Paula Regina Kuser Falcão

Embrapa Informática Agropecuária
Campinas, SP
2007

Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)
Av. André Tosello, 209
Cidade Universitária "Zeferino Vaz" Barão Geraldo
Caixa Postal 6041
13083-970 - Campinas, SP
Telefone (19) 3789-5743 - Fax (19) 3289-9594
URL: http://www.cnptia.embrapa.br
e-mail: sac@cnptia.embrapa.br

Supervisor editorial: *Ivanilde Dispato*
Normalização bibliográfica: *Marcia Izabel Fugisawa Souza*
Editoração eletrônica: *Área de Comunicação e Negócios (ACN)*

1ª. edição on-line - 2007

Cálculo analítico de parâmetros estruturais de proteínas usando a teoria Alpha Shapes / George Barreto Pereira Bezerra...[et al.]. — Campinas : Embrapa Informática Agropecuária, 2007.
48 p. : il. — (Boletim de Pesquisa e Desenvolvimento / Embrapa Informática Agropecuária; 17).

ISSN 1677-9266

1. Bioinformática. 2. Geometria computacional. 3. Triangulação de Delaunay. 4. Teoria de Alpha Shapes. 5. Estrutura de proteína. I. Bezerra, George Barreto Pereira. II. Série.

CDD - 570.285 (21st. Ed.)

Sumário

Cálculo Analítico de Parâmetros Estruturais de Proteínas usando a Teoria Alpha Shapes

George Barreto Perreira Bezerra[1]
Michel Eduardo Beleza Yamagishi[2]
Fábio Danilo Vieira[3]
Edgard Henrique dos Santos[4]
Marcelo Gonçalves Narciso[5]
Paula Regina Kuser Falcão[6]

Resumo

O modelo matemático da estrutura protéica é fundamental não só para a visualização, mas, principalmente, para o cálculo de parâmetros estruturais como área da superfície e volume. É possível usar a geometria computacional para efetuar o cálculo analítico de tais parâmetros. Neste trabalho, usando Alpha-shapes, um algoritmo para o cálculo analítico de área da superfície e volume de estruturas protéicas do Protein Data Bank (PBD) é descrito. Esta implementação supera as demais, pois consegue calcular tais parâmetros para estruturas protéicas onde as demais implementações falham.

Palavras-chaves: Geometria computacional, triangulação de Delaunay, alpha-shapes, estrutura de proteína.

[1] Doutorando em Engenharia Elétrica da Universidade de Novo México. Albuquerque, USA, NM 87131-0001.
[2] Doutor em Matemática Aplicada, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo -13083-970 - Campinas, SP.
[3] Bacharel em Tecnologia da Informação, Analista da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP.
[4] Bacharel em Ciência da Computação, Analista da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP.
[5] Doutor em Computação, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP.
[6] Doutora em Cristalografia de Proteínas, Pesquisadora da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP.

Analytical Calculation of Protein Strutural Parameters using Alpha Shapes Theory

Abstract

The mathematical model of protein structure is essential not only for visualization purpose, but specially for the computation of structural parameters such as superficial area and volume. It is possible to apply computational geometry to perform the analytical calculation of these parameters. In this work, using Alpha-shapes, an algorithm to calculate analytically superficial area and volume of protein structures deposited in the Protein Data Bank (PDB) is described. This implementation outperforms other implementations, because it succeeds in calculating those parameters of protein structures which other implementations fail.

Index Terms: Computational geometry, Delaunay triangulation, alpha-shapes, protein structure

Introdução

O DNA é uma macromolécula que contém as informações necessárias para o desenvolvimento e funcionamento de todos os seres vivos. Essas informações são armazenadas na forma de seqüências de nucleotídeos, tendo trechos específicos, chamados genericamente de genes, que são transcritos em mRNA e depois traduzidos em proteínas, que por sua vez exercerão diversas funções no organismo. Esse processo de síntese de proteína a partir das informações codificadas em um gene é chamado expressão gênica (Albert et al., 2002).

Cada organismo possui uma grande quantidade de proteínas diferentes, e cada uma delas possui uma estrutura específica à qual está associada uma determinada função. A grande maioria das reações metabólicas ocorre através de encaixes do tipo chave-fechadura entre moléculas protéicas e algum substrato, que pode ser outra proteína, e, por esse motivo, a sua estrutura é tão importante. Se uma mutação num gene altera minimamente a parte funcional da estrutura da proteína, a deformidade causada pode inviabilizar todas as reações nas quais ela estava envolvida, inutilizando assim a proteína. Diversas doenças conhecidas são causadas por eventos deste tipo. É o caso da anemia falciforme, que ocorre quando há mudança de um aminoácido na proteína beta-globina; fibrose cística, na qual a proteína reguladora de condutância transmembrana sofre uma alteração causando transporte anormal de íons de cloro; catarata, causada por uma alteração na proteína gama-cristalina; e fenilcetonúria, um erro no metabolismo que causa retardo mental devido ao mal funcionamento da enzima fenilanina hidroxilase.

Estudar as proteínas é, portanto, uma etapa fundamental para o entendimento dos sistemas e processos biológicos. Esse estudo envolve associar funções às proteínas, classificá-las, encontrar relações entre elas, entender suas interações e identificar padrões. Entretanto, para que isso seja feito é necessário desvendar a estrutura tridimensional das proteínas, identificando regiões funcionais e não-funcionais, investigando propriedades físico-químicas, estudando interações locais de cadeias protéicas e os motivos que levam a sua conformação final. Com essas informações os cientistas podem, por exemplo, prever interações com outras moléculas, atribuir funções a novas proteínas através de similaridade de seqüência ou estrutural com proteínas já conhecidas, e por fim, entender melhor a causa de doenças como o câncer, elaborar novas drogas e terapias, e projetar ferramentas para realizar diagnósticos.

Desvendar a estrutura de uma proteína é um processo bastante complicado. Inicialmente, é preciso obter as coordenadas tridimensionais (3D) de todos os seus átomos. Isto é feito atualmente através de técnicas como cristalografia de raios X e ressonância magnética nuclear, mas essas técnicas não funcionam para todas as proteínas. Por este motivo, muitas moléculas de grande interesse para a ciência ainda não tiveram suas coordenadas atômicas determinadas. Sobre os dados das coordenadas atômicas são aplicados algoritmos matemáticos e estatísticos, com o

objetivo de gerar uma grande quantidade de parâmetros relacionados à estrutura da molécula. Esses parâmetros são utilizados para se compreender melhor a relação entre seqüência-estrutura-função das proteínas. A comunidade científica pode acessá-los gratuitamente através da internet, bastando utilizar uma interface gráfica amigável, como o STING (Neshich et al., 2003) ou o Java Protein Dossier (JPD) (Neshich et al., 2004).

Em todo esse processo, a bioinformática tem um papel de grande responsabilidade, principalmente na elaboração de ferramentas da matemática aplicada e de computação para a geração dos parâmetros e na criação e manutenção de uma interface acessível aos pesquisadores. Sob esta mesma perspectiva, este trabalho aborda o uso de conceitos e técnicas de geometria computacional e matemática aplicada na implementação de algoritmos computacionais para o cálculo analítico de parâmetros estruturais de proteínas. Os algoritmos propostos se baseiam em conceitos geométricos como diagrama de Voronoi, triangulação de Delaunay e teoria de Alpha Shapes. As implementações envolvem um código para visualização de interface entre cadeias protéicas e algoritmos para o cálculo analítico do volume e área da superfície da proteína.

Material e Métodos

As proteínas são as moléculas responsáveis pelo metabolismo e estrutura dos organismos vivos. Elas atuam em funções como catálise enzimática, transporte e armazenamento, movimento coordenado, sustentação mecânica, ação imunológica, e geração e transmissão de impulsos nervosos – eventos que ocorrem dentro e fora das células dos organismos. Entender como os processos biológicos ocorrem em nível celular, implica, particularmente, em entender o papel funcional das diversas proteínas envolvidas nesses processos. Como a função de uma proteína está geralmente associada à sua conformação tridimensional (Branden & Tooze, 1999), diversas pesquisas voltadas à compreensão da funcionalidade dessas macromoléculas estão associadas ao estudo da sua estrutura.

As proteínas são compostas por uma cadeia de aminoácidos conectados de forma seqüencial, e a sua estrutura pode ser discutida em quatro níveis:

- Estrutura Primária: corresponde à própria seqüência linear de aminoácidos.
- Estrutura Secundária: são conformações locais tomadas a partir de interações de curta distância entre aminoácidos.
- Estrutura Terciária: é a estrutura tridimensional, formada pelo conjunto global de interações entre os aminoácidos.
- Estrutura Quaternária: é a estrutura obtida através de interações entre cadeias protéicas.

Acredita-se que a estrutura primária de uma proteína determina todas as outras (Epstein et al., 1963), mas até o momento ainda não é possível predizer a estrutura tridimensional das proteínas baseado, exclusivamente, na seqüência linear de aminoácidos que as compõem, embora existam inúmeras iniciativas neste sentido. A razão é que as interações que levam a proteína a se dobrar várias vezes até adquirir uma conformação estável ainda não são bem compreendidas. Além disso, o número de possíveis estruturas que ela pode assumir é imenso, e determinar dentre elas a mais estável é um problema de natureza combinatorial impossível de ser resolvido na prática (Levinthal, 1968).

A comunidade científica conta com 43.459 estruturas de proteínas expressas em termos de coordenadas atômicas (dados de 16.05.2007), depositadas no PDB (Berman et al., 2000). Esse número tem aumentado bastante nos últimos anos, apresentando uma curva de crescimento exponencial (Fig. 1).

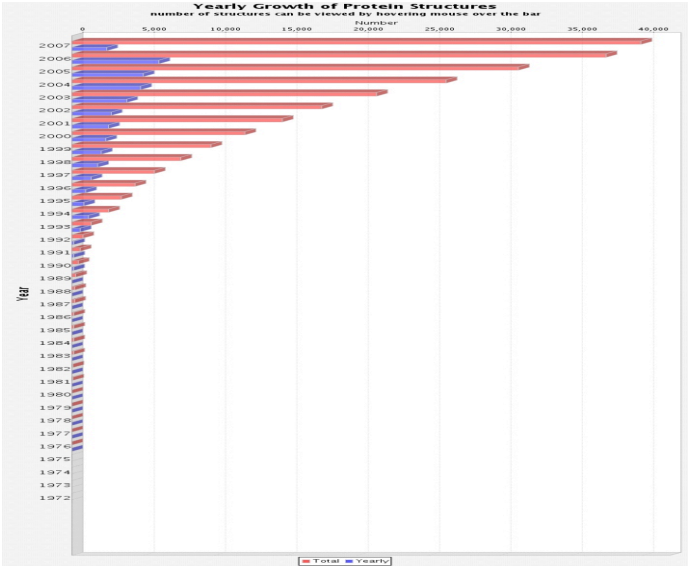


Fig. 1. Crescimento exponencial do número de estruturas disponíveis no PDB.
Fonte: www.pdb.org.

Como já mencionado, essas estruturas são determinadas através de técnicas de cristalografia de raios X e ressonância magnética nuclear. Os dados correspondem à posição tridimensional do centro de cada átomo da proteína, isto é, as suas coordenadas nos eixos cartesianos *x-y-z.*

Com estas informações é possível criar um modelo geométrico da proteína para analisar a sua estrutura com a ajuda de ferramentas matemáticas. Programas de computador podem ser utilizados para calcular parâmetros estruturais que determinam uma variedade de propriedades físicas e químicas.

Para modelar geometricamente uma proteína, os diagramas de preenchimento de espaço (*space-filling diagrams*) ou envelope de van der Waals (Lee & Richards, 1971) são comumente usados como representação. Nesses diagramas, cada átomo é representado por uma bola esférica com raio de van der Waals. A molécula como um todo se torna uma composição de bolas que se sobrepõem assumindo a sua conformação final no espaço tridimensional. A Fig. 2 ilustra a representação de van der Waals de uma molécula.

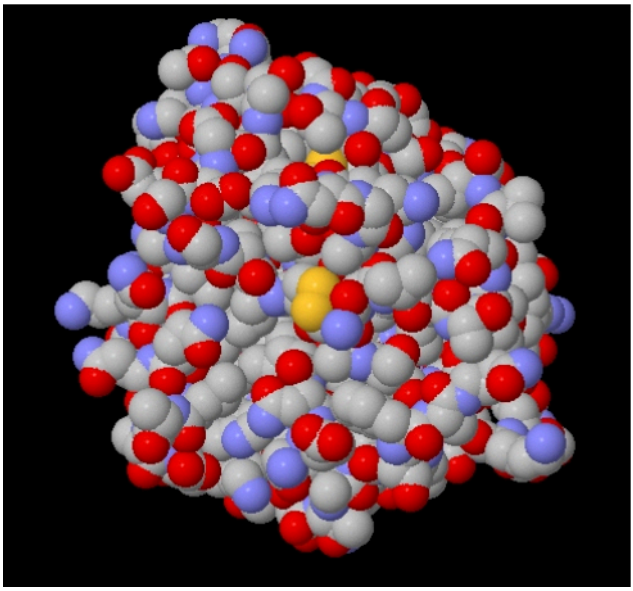


Fig. 2. Envelope de van der Waals de uma molécula. Cada cor representa um tipo diferente de átomo.

A superfície determinada por este modelo é chamada superfície de van der Waals (VW). Em Lee & Richards (1971) mais dois modelos de superfície foram definidos: a superfície acessível ao solvente (SA - Solvent Accessible surface) e superfície molecular (MS - Molecular Surface). A superfície acessível ao solvente é determinada pelo centro do solvente, modelado como uma esfera sólida rolando sobre a molécula. A superfície molecular é formada pela parte da mesma esfera que mais se aproxima da molécula.

Esses modelos são extremamente úteis à biologia molecular pois definem parâmetros estruturais das proteínas sem ambigüidades. A Fig. 3 faz uma comparação entre os três modelos.

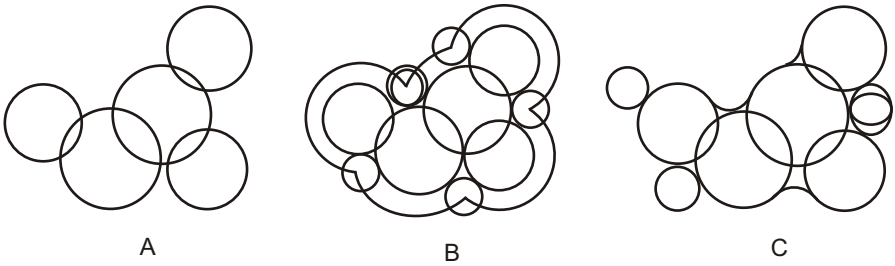


Fig. 3. Modelos de superfície molecular (A) superície de van der Waals (VW) (B) superfície acessível ao solvente (SA) e superfície molecular (MS).

A determinação dos parâmetros estruturais gerados a partir desses modelos são muito importantes para proteômica funcional. A área da superfície, o volume molecular (definidos para os diferentes modelos) são quantidades que determinam várias propriedades das proteínas, como estabilidade da conformação tridimensional, solubilidade, reconhecimento molecular e catalisação enzimática. Além disso, cavidades existentes nas proteínas também contribuem para a sua funcionalidade, influenciando na estabilidade termodinâmica, permitindo mudanças conformacionais e acomodando pequenas moléculas. Calcular área e volume dessas cavidades também é muito importante para determinar a funcionalidade das proteínas.

Diversos métodos já foram desenvolvidos para o cálculo desses parâmetros. A maioria deles é baseado em aproximações (Shrake & Rupley, 1973; Richards, 1974; Richmond & Richards, 1978; Alden & Kim, 1979; Wodak & Janin, 1980; Muller, 1983; Pavlov & Fedorov, 1983; Pascual-Ahuir & Silla, 1990; Wang & Levinthal, 1991; Grand & Mertz Junior, 1993) e outros em cálculos analíticos (Kundrot et al., 1991; Connoly, 1983, 1985a, 1985b; Richmond, 1984; Gibson & Sheraga, 1987, 1988; Perrot et al., 1992). Métodos numéricos, baseados em aproximações, envolvem uma discretização da superfície em um grande número de pontos, o que gera imprecisões nos cálculos. Para aumentar a precisão do método é necessário um número maior de pontos, o que geralmente aumenta muito o custo computacional. Métodos analíticos encontram o resultado “exato” para os parâmetros, representando os átomos como bolas esféricas. Problemas com métodos analíticos envolvem o tratamento de situações especiais como o cálculo de interseções de cinco ou mais átomos.

Dado um conjunto S de n pontos num espaço $\Re^d$, o diagrama de Voronoi (Richards, 1974; Finney, 1975; Gellatly & Finney, 1982) divide este espaço em n polígonos convexos, chamados regiões de Voronoi, uma para cada um dos pontos de S. Uma região de Voronoi V, definida para um ponto $p \in S$, é a parte do espaço que é mais próxima de p de que qualquer outro ponto de S. Assim, a distância entre um ponto q qualquer de V a p é menor ou igual à distância entre q e qualquer outro ponto de S. A Fig. 4 mostra um exemplo do diagrama de Voronoi para um conjunto de 13 pontos $\Re^2$. Note que embora a figura esteja limitada por bordas, o diagrama de Voronoi se estende ao infinito para os pontos mais externos.

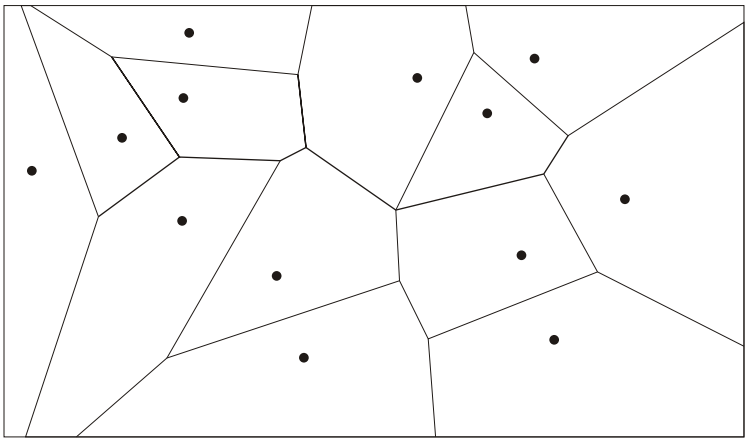


Fig. 4. Diagrama de Voronoi para um dado conjunto de pontos.

O diagrama de Voronoi possui propriedades inerentes que se mostram atraentes para o tratamento de problemas envolvendo átomos e moléculas. De fato, esta metodologia possui um vasto campo de aplicações em áreas da ciência como biologia e química (David & David, 1982; Aurenhammer, 1991). Para se construir um diagrama envolvendo átomos, basta considerar o centro de cada átomo como um ponto em $\Re^3$ definindo uma região de Voronoi. Se todos os átomos possuem o mesmo raio de van der Waals, o plano que separa dois átomos de mesmo tamanho será o plano bissetor, no qual cada ponto é eqüidistante ao centro dos dois átomos. Assim, o espaço interno de uma macromolécula pode ser totalmente decomposto em polígonos que armazenam todas as informações de vizinhança entre seus átomos.

Os tipos de átomos encontrados em uma mesma macromolécula, como uma proteína, costumam ser bastante variados. Átomos diferentes possuem raios diferentes, portanto é necessário determinar uma maneira de construir um diagrama de Voronoi para uma molécula que leve em consideração a diferença no tamanho de seus átomos.

Em casos como esse, o procedimento adotado é utilizar uma outra medida de distância, a distância ponderada, ao invés da distância Euclidiana. A distância ponderada de um átomo (ou de uma esfera qualquer) a um ponto é definida como o quadrado do comprimento do segmento formado pela linha tangente à superfície do átomo e que passa pelo ponto. Formalmente, a distância ponderada entre um átomo p e um ponto x é definida como $\pi_p(x)^2 = d(x,p)^2 - w_p^2$, onde $d(x,p)$ é a distância entre x e o centro de p e w_p é o raio de p. A Fig. 5 mostra em detalhes a distância ponderada.

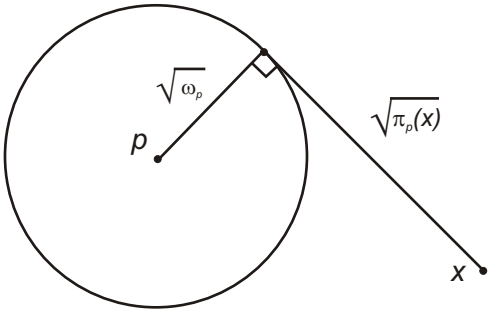


Fig. 5. Distância ponderada entre uma esfera e um ponto.

O plano que divide dois átomos será chamado agora de *plano radical* (Fig. 6), no qual todos os pontos possuem segmentos tangentes às duas esferas com o mesmo comprimento. A Fig. 6 ilustra esta situação para um caso em duas dimensões.

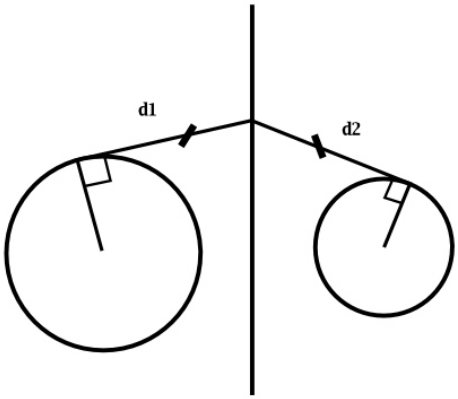


Fig. 6. Plano radical separando duas esferas de raios diferentes.

Repare que o comprimento dos segmentos tangentes à superfície dos discos que vão até um ponto qualquer do plano radical é sempre igual. Observe também que o plano radical é sempre paralelo ao plano bissetor.

Utilizando esta medida de distância é possível manter a analogia anterior, construindo um diagrama de Voronoi ponderado que armazena também informações sobre os raios dos átomos. Ele possui as mesmas propriedades do diagrama não-ponderado. A Fig. 7, apresenta uma ilustração do diagrama ponderado construído em duas dimensões para uma molécula bidimensional com átomos de raios diferentes.

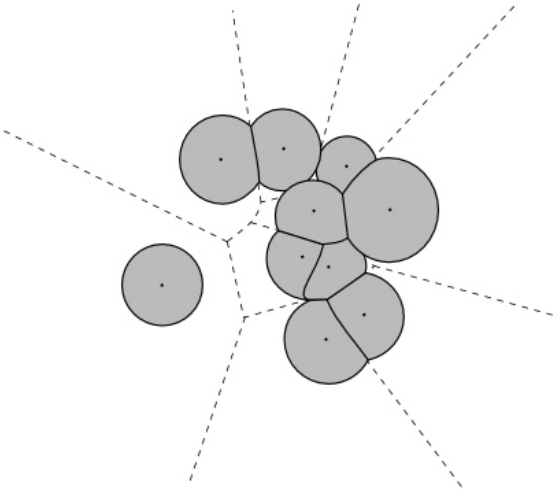


Fig. 7. Diagrama de Voronoi ponderado para uma molécula bidimensional.

O diagrama de Voronoi separa os discos em pequenas regiões convexas, ponderadas pelos raios. Esta representação é particularmente importante para moléculas, pois ela decompõe totalmente o seu interior sem apresentar superposições redundantes.

Para um caso geral em terceira dimensão, duas regiões de Voronoi não possuem interseção, ou então compartilham um mesmo plano do diagrama, três regiões não possuem interseção ou se encontram em uma mesma aresta e quatro regiões não possuem pontos em comum ou então se encontram em apenas um ponto. Interseções de cinco ou mais esferas não ocorrem sob esta representação, embora se saiba que interseções de mais de quatro átomos ocorrem em proteínas reais na natureza.

Esses diagramas têm sido bastante utilizados em bioinformática para a definição teórica de parâmetros estruturais das proteínas, como área da

superfície e volume, assim como para a determinação prática de seus valores. Diversas técnicas foram desenvolvidas para estes cálculos (Gertein & Richards, 2006), mas os diagramas de Voronoi apresentam alguns problemas que dificultam o seu uso em termos de tratamento computacional. Uma delas é que embora os átomos em moléculas reais ocupem uma região finita do espaço, as regiões de Voronoi podem se estender infinitamente. Os átomos dispostos na superfície da molécula definem regiões infinitas, o que dificulta muito a representação dos diagramas em um computador, e não permite a aplicação direta de fórmulas para calcular os parâmetros. Nestes casos é necessário utilizar artifícios para realizar os cálculos, como, por exemplo, criar moléculas esféricas de solvente hipotéticas ao redor da proteína com o objetivo de definir planos radicais que limitem o espaço ocupado pelas regiões de Voronoi representando a superfície da macromolécula. Outro problema desta abordagem é que o diagrama é composto de um conjunto de vértices, segmentos e planos que não pertencem ao conjunto de dados de entrada. É uma quantidade de informação adicional considerável a ser armazenada, aumentando assim a demanda de memória computacional. Além disso, calcular esse novo conjunto de vértices pode ser bastante complicado.

É possível, entretanto, representar as mesmas informações contidas no diagrama de Voronoi de uma maneira mais adequada. A Triangulação de Delaunay armazena o mesmo conteúdo combinatório dos diagramas, mas não apresenta os mesmos problemas de representação num computador. Conforme descrito a seguir, existe uma dualidade entre as duas representações e a partir do diagrama de Voronoi é possível construir a triangulação de Delaunay e vice-versa.

Para definir a triangulação de Delaunay de um conjunto de pontos, é preciso antes introduzir o conceito de *convex hull*. Um subconjunto $S \subseteq \Re^d$ é considerado convexo se para dois pontos quaisquer *p* e *q* de *S* o segmento de reta ligando *p* a *q* está contido em *S*. O *convex hull* de um conjunto *S* é o menor conjunto convexo que contém *S*. O *convex hull* de um conjunto de pontos *P* é um polígono convexo com vértices em *P*. Já um ponto de *P* que corresponde a um vértice do *convex hull* é chamado ponto extremo (com relação a *P*).

Uma maneira intuitiva de apresentar o conceito de *convex hull* é a seguinte: imagine um conjunto de pontos em três dimensões fixos no espaço. Quando se envolve este conjunto de pontos com um filme plástico e apertar-se bem, o sólido resultante (formado pelo filme plástico) será o *convex hull* deste conjunto de pontos. A Fig. 8 mostra um exemplo de *convex hull* em duas dimensões para um dado conjunto de pontos.

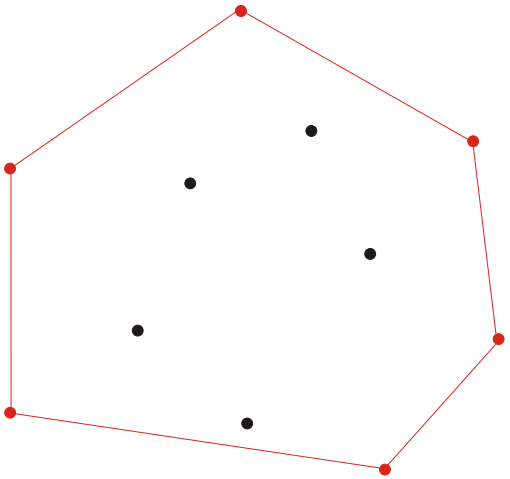


Fig. 8. Ilustração do *convex hull* para um conjunto de pontos em duas dimensões.

Para um caso geral em $\Re^3$, a triangulação de Delaunay (Delaunay, 1934) de um conjunto de pontos *P* consiste na decomposição do *convex hull* em tetraedros (em triângulos no caso bidimensional), sendo os pontos de *P* os vértices dos tetraedros, de tal forma que a esfera que circunscreve cada tetraedro não contém nenhum outro ponto de *P*. A triangulação de Delaunay é sempre única, quando se considera os pontos de *P* em uma *posição geral*, isto é, não há quatro ou mais pontos em um mesmo plano nem cinco ou mais pontos em uma mesma esfera.

O *convex hull* é então preenchido de forma que as arestas de cada tetraedro não cruzam nem interceptam triângulos, a não ser que eles compartilhem um mesmo vértice. Nesta decomposição, os tetraedros estão dispostos de tal forma que dois deles podem compartilhar um vértice, uma aresta ou até um triângulo; caso contrário, não há interseção. Este conjunto de vértices, arestas, triângulos e tetraedros é chamado *complexo simplicial*. Um complexo simplicial é definido para uma triangulação qualquer, e não apenas para a triangulação de Delaunay. Este conceito será melhor discutido mais adiante.

A Fig. 9 mostra a decomposição de um *convex hull*, no caso de um conjunto de pontos bidimensionais, em uma triangulação de Delaunay.

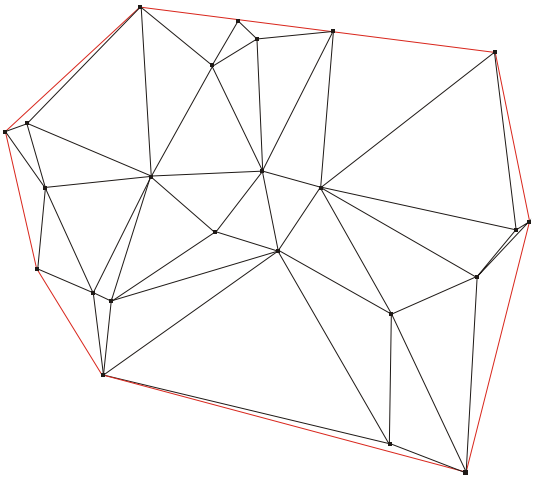


Fig. 9. Decomposição do *convex hull* em uma triangulação de Delaunay.

A triangulação de Delaunay é formalmente definida para pontos, ou esferas de mesmo raio. Esta propriedade torna a triangulação inadequada para se trabalhar com modelagem de macromoléculas. Assim como no caso do diagrama de Voronoi, é necessário construir uma triangulação que leve em consideração os diferentes raios dos átomos. Utilizando o conceito de distância ponderada é possível chegar a uma triangulação que possui esta propriedade. Esta triangulação é chamada *triangulação regular*, que corresponde a um caso genérico da triangulação de Delaunay.

Como dito anteriormente, existe uma dualidade entre o diagrama de Voronoi e a triangulação de Delaunay. Esta dualidade é a mesma existente entre o diagrama ponderado e a triangulação regular. Em $\Re^3$, um vértice da triangulação de Delaunay corresponde a uma região de Voronoi, uma aresta corresponde a um plano no diagrama, um triângulo corresponde a uma aresta e um tetraedro a um vértice de Voronoi. Da mesma forma, em $\Re^2$ cada vértice, aresta e triângulo de Delaunay representa uma região, uma aresta e um vértice de Voronoi. A dualidade entre as duas representações é evidenciada na Fig. 10. A Fig. apresenta uma superposição entre o diagrama de Voronoi e a triangulação de Delaunay (ambos ponderados), construídos sobre o mesmo conjunto de discos da Fig. 7.

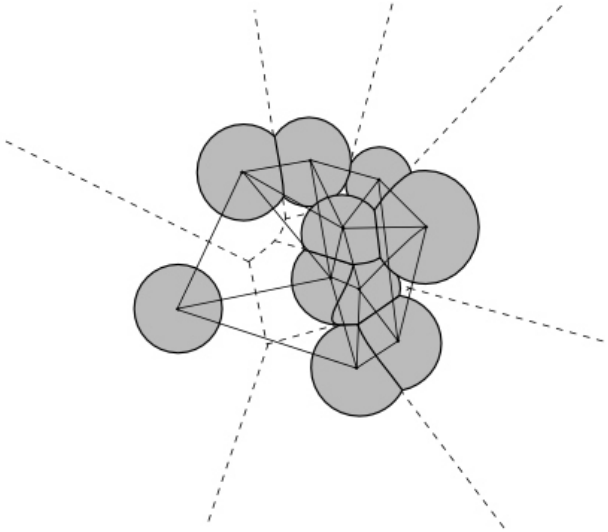


Fig. 10. Sobreposição da triangulação de Delaunay e o diagrama de Voronoi, ambos ponderados.

As duas representações contêm a mesma informação sobre os átomos, e que através de uma é possível chegar à outra. Esta dualidade representa uma propriedade muito interessante para a geometria computacional. Ela pode ser utilizada, por exemplo, para facilitar o cálculo dos parâmetros estruturais de proteínas. Isto porque é muito mais fácil elaborar algoritmos robustos para construir a triangulação de Delaunay que para construção do diagrama de Voronoi. A principal razão é que as triangulações de Delaunay não inserem informações geométricas adicionais e todas as arestas, triângulos e tetraedros podem ser armazenados como conjuntos de dois, três e quatro vértices, respectivamente, o que facilita a sua representação em um computador. Já o diagrama de Voronoi contém vértices que não fazem parte do conjunto de dados de entrada, dificultando uma representação computacional e exigindo uma capacidade adicional de memória. Outra grande vantagem da triangulação é que o espaço onde ela está inserida é sempre limitado, não se estende ao infinito. Algoritmos para a construção da triangulação de Delaunay são descritos em Delaunay (1934) e Chazelle et al. (2001).

Uma triangulação qualquer em três dimensões é constituída de uma composição de unidades básicas, isto é, vértices, arestas, triângulos e tetraedros, chamadas *simplexos*. O conjunto de todos os *simplexos* de uma triangulação é chamado *complexo simplicia*l, e para o caso específico de uma triangulação de Delaunay, o complexo simplicial é chamado *complexo de Delaunay.*

Uma notação bastante utilizada é chamar um vértice da triangulação de *0-simplex*, uma aresta de *1-simplex*, um triângulo de *2-simplex*, e um tetraedro de *3-simplex*. O número inteiro corresponde à dimensão atribuída a cada um dos simplexos. A Fig. 11 apresenta cada um dos simplexos para $\Re^3$ e uma coleção de simplexos arranjados de forma a constituir um

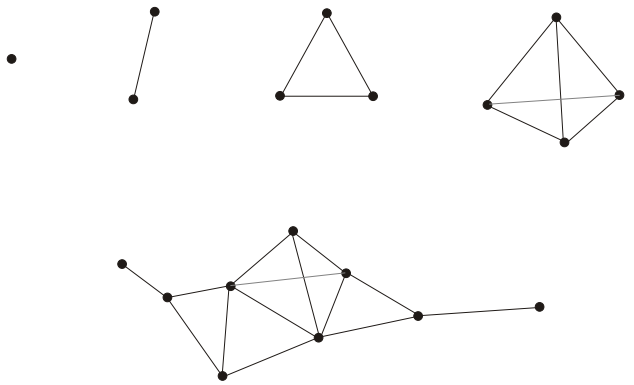


Fig. 11. Em cima, da esquerda para a direita, os quatro tipos de simplexos num espaço tridimensional: vértice, aresta, triângulo e tetraedro. Abaixo, um complexo simplicial formado pela combinação de vários simplexos.

Um simplex σ_T é definido para um conjunto de T pontos, tal que a dimensão do simplex é determinada pelo número de pontos em T menos um, ou seja, o número de pontos em T é $|T| = k + 1$, e a dimensão k do k-simplex σ_T é $k = |T| - 1$, com $0 \leq k$. Formalmente, um complexo simplicial denotado por C é uma coleção de k-simplexos que satisfaz as seguintes propriedades:

(1) Se $\sigma_T \in C$ então $\sigma_{T'} \in C$ para todo $T' \subseteq T$. Em outras palavras, para todo simplex σ_T, C contém todas as faces de σ_T.

(2) Se σ_T e $\sigma_{T'} \in C$, então ou $\sigma_T \cap \sigma_{T'} = \varnothing$ ou $\sigma_T \cap \sigma_{T'} = \sigma_{T \cap T'} =$ *convex hull* de $(T \cap T')$. Isto quer dizer que a interseção de dois simplexos ou é vazia ou é uma face de ambos.

A maior dimensão de um simplex $\sigma_T \in C$ é chamada dimensão de C. Um subconjunto $C' \subseteq C$ é chamado de *subcomplexo* de C se ele também é um complexo simplicial.

A seguir descreve-se o que são os alpha shapes e como eles podem ser utilizados para calcular os parâmetros estruturais das macromoléculas. Embora não seja difícil de entender intuitivamente a definição de alpha shapes, é necessária uma formulação matemática rigorosa, com a introdução de uma série de conceitos e particularidades geométricas, a fim de determinar uma metodologia para o cálculo dessas estruturas. Um algoritmo para a construção dos alpha shapes também será apresentado e discutido neste trabalho.

O *alpha shape* pode ser explicado simplificadamente da seguinte forma: considere um conjunto de pontos fixos no espaço $\Re^3$. Para saber a "forma" geométrica tridimensional determinada por estes pontos para um dado α (isto é, o *alpha shape*), basta imaginar uma esfera de raio fixo α que se movimenta ao redor deles, podendo encostar nos pontos, mas não atravessá-los. O *alpha shape* será então o sólido formado por todos os pontos que a esfera de raio α consegue atingir, considerando inclusive o interior do conjunto de pontos. Um exemplo em duas dimensões do *alpha shape* é mostrado na Fig. 12. A Fig. mostra uma circunferência de raio fixo se movimentando em volta dos pontos, inclusive pelo seu interior (se há espaço para ela), determinando um formato para o conjunto de pontos. Quando se diminui o valor de α, isto é, se reduz o raio da esfera, ela atingirá mais pontos, dando ao conjunto um novo formato.

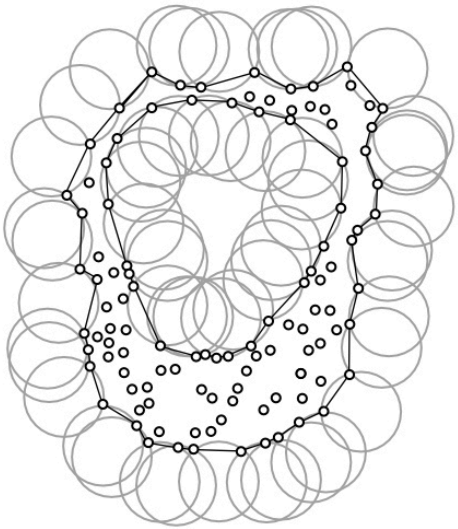


Fig. 12. Alpha shape de um conjunto de pontos em duas dimensões, determinado por uma esfera de raio α fixo.

Através da Fig. 12 é possível ver que cada valor de α determina um formato, ou "shape", diferente para os pontos. O conjunto de todos os formatos geométricos diferentes que o conjunto de pontos pode assumir é chamado de *alpha shapes*.

É possível ainda detalhar um pouco mais a idéia de α-shapes. Considere novamente um conjunto de pontos qualquer num espaço tridimensional. Imagine agora que em volta de cada ponto existe uma esfera cujo raio é variável e igual a um parâmetro α. Cada esfera pode crescer apenas dentro de sua região de Voronoi, assim como os discos das Fig. 7 e 10. No momento que uma esfera atinge o limite de sua região de Voronoi, isto é, no momento em que duas esferas se encontram definindo um plano da região de Voronoi, surge um simplex representando a interseção entre as duas esferas. Inicialmente, α é igual a zero. As esferas coincidem com os pontos do conjunto e os únicos simplexos definidos são os vértices. À medida que α começa a aumentar, as esferas também crescem até ocorrer uma eventual superposição de duas esferas. No momento em que uma superposição acontece, surge uma aresta (do complexo de Delaunay) representando a interseção das esferas. Este instante em que a aresta surge é marcado cronologicamente no complexo de Delaunay, com relação ao valor de α que lhe deu origem. A aresta entra então para uma seqüência cronológica de simplexos, na qual até agora só havia vértices. Quando as esferas crescem o suficiente até que três delas se encontram, surge um triângulo do complexo de Delaunay, marcado pelo α que corresponde ao raio das esferas. Este triângulo também é inserido na seqüência de simplexos. O mesmo é feito para um tetraedro, definido no momento em que quatro esferas se superpõem. Este processo de aumentar o valor de α ocorre até que as esferas estejam grandes o suficiente a ponto de definir todos os simplexos do complexo de Delaunay. O resultado final é uma seqüência de simplexos organizada cronologicamente de acordo com o valor de α, chamada *filtro*. Se escolhe uma posição qualquer do filtro, todos os simplexos definidos até aquele ponto formam juntos um complexo, que é na verdade um subcomplexo do complexo de Delaunay. Observe, portanto, que os α-shapes e as representações de Delaunay e Voronoi apresentam conceitos fortemente inter-relacionados.

A descrição apresentada se refere a pontos ou então a esferas de mesmo raio no espaço. Para o caso de átomos diferentes é preciso definir uma idéia de α-shapes que seja ponderada pelos raios dos átomos. O *weighted alpha shapes* (Edelsbrunner, 1992), ou *alpha shapes ponderado*, pode ser obtido através de uma triangulação regular, ao invés de uma triangulação de Delaunay, e as regiões de crescimento dos átomos são definidas pelo diagrama de Voronoi ponderado. O parâmetro α ainda determina o crescimento ou encolhimento das esferas, mas será ponderado pelo raio de cada átomo através da seguinte equação:

$$r_\alpha = \sqrt{w_0^2 + \alpha^2} \quad (1)$$

Onde w_0 é o raio de van der Waals do átomo. Para α igual a zero tem-se então o tamanho real da molécula.

Assim como no caso anterior, através do aumento do valor de α é possível chegar também a uma coleção ordenada de simplexos, isto é, o filtro. Mais uma vez, uma posição qualquer do filtro corresponde a um determinado valor de α definindo o raio das esferas. Todos os simplexos do filtro já definidos até este valor de α formarão um complexo simplicial. O complexo simplicial para um dado α é chamado *complexo alpha* (Mücke, 1993; Edelsbrunner & Mücke, 1994). O α-shape é a parte do espaço coberta por todos os simplexos do complexo-α.

A Fig. 13 ilustra o efeito do crescimento ponderado dos átomos sobre a triangulação. A molécula em duas dimensões é a mesma das Fig. 7 e 10. Inicialmente α é igual a zero e os átomos possuem seu tamanho original. Há apenas alguns simplexos, representando os centros dos átomos e algumas interseções que ocorrem entre eles na molécula real. Na medida em que α aumenta, as esferas começam a se sobrepor, compondo um complexo simplicial cada vez maior. No último estágio mostrado na Fig. 12, as esferas estão bastante grandes, e quase todos os simplexos do complexo de Delaunay já estão presentes na triangulação. Observe que os simplexos definidos para um valor pequeno de α também estão presentes no complexo definido por um valor alto de α. Complexos para alphas pequenos são subcomplexos de complexos para alphas grandes, e ambos são subcomplexos do complexo de Delaunay.

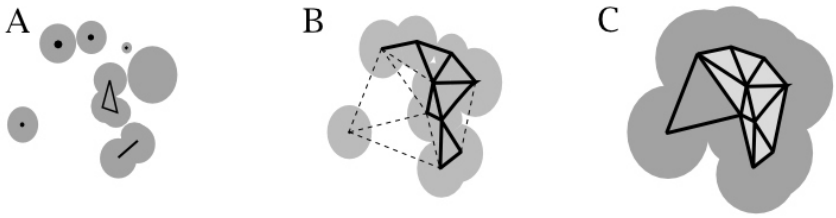


Fig. 13. Efeito do crescimento de α sobre o complexo.

Uma propriedade interessante dos α-shapes é que as mudanças ocorrem em passos discretos, com a introdução de cada novo simplex, independentemente de quanto α muda de um passo a outro. Considere por exemplo a cavidade que existe no meio da molécula da Fig. 13 na fase B.

Esta cavidade corresponde a um grande triângulo no complexo-α. Quando α assume um valor maior, como em C, o triângulo surge e a cavidade desaparece. O mais interessante aqui é que assim como o surgimento de cada união de discos está presente no complexo-α, as cavidades existentes na molécula podem ser vistas como uma cavidade no complexo-α. Desta maneira, é possível identificar todas as cavidades de uma proteína, independente do seu tamanho. Além disso, átomos presentes na superfície da molécula podem ser identificados com precisão. Outra vantagem dos α-shapes é que a correspondência entre o complexo-α e a molécula permite que se obtenha a sua área e volume diretamente através do complexo-α, sem necessitar construir explicitamente um modelo geométrico para a molécula.

Como mencionado anteriormente, os complexos-α podem ser utilizados para o cálculo de parâmetros estruturais como volume e área das macromoléculas e de suas cavidades. São descritas a seguir algumas metodologias que definem como os α-shapes podem ser utilizados com estas finalidades.

O cálculo do volume e área de macromoléculas é um assunto bastante abordado na literatura. O problema é difícil, pois os átomos correspondem a bolas esféricas de raios diferentes e que se sobrepõem. Se não houvesse interseção entre os átomos, o problema seria facilmente resolvido somando-se as áreas e volumes individuais de cada átomo. Infelizmente, em moléculas reais as interseções são freqüentes e costumam acontecer até para mais de quatro átomos.

Existe, no entanto, uma abordagem conceitualmente simples para tratar o problema. Trata-se do princípio da *inclusão-exclusão*, no qual se dois átomos se sobrepõem, subtrai-se a interseção, se três átomos se sobrepõem, primeiro retira-se as interseções dois a dois e adiciona-se a interseção tripla, etc. O mesmo processo ocorre para interseções de quatro ou mais átomos. Embora o princípio da inclusão-exclusão seja intuitivo, simples e capaz de resolver o problema em questão, existem alguns fatores que dificultam a sua aplicação direta sobre as coordenadas 3D das macromoléculas. Através dos complexos-α, no entanto, é possível eliminar os problemas do princípio de inclusão-exclusão, levando a um método eficiente para o cálculo de área e volume chamado *método da inclusão-exclusão direta* (Liang et al., 1998a).

Um dos problemas do princípio da inclusão-exclusão é que para aplicar o método, é necessário ter todas as informações de vizinhança dos átomos, e no caso das macromoléculas, em que o número de átomos é muito grande, obter todas as interseções nelas existentes se torna um problema

combinatorial muito custoso computacionalmente. Com os complexos-α essas informações de relação de vizinhança entre os átomos são geradas automaticamente para toda a molécula. Além disso, todas as interseções de cinco ou mais átomos são reduzidas a combinações de interseções de quatro ou menos esferas. Isto é uma grande vantagem, pois o cálculo de interseções de cinco ou mais esferas é demasiadamente complicado. A prova para estas reduções é dada em (Kratky, 1981) para duas dimensões. Entretanto, esses resultados têm sido amplamente aplicados em casos de três dimensões mesmo sem uma prova explícita.

Outro aspecto negativo da regra de inclusão-exclusão é que a fórmula do cálculo de área e volume geralmente possui vários termos redundantes, o que complica o cálculo desnecessariamente. Como conseqüência, o tempo de processamento se torna muito elevado, podendo inviabilizar a utilidade prática do método. A inclusão-exclusão direta, no entanto, consegue evitar esses termos redundantes, levando a uma fórmula muito mais enxuta. Para entender melhor a vantagem do uso dos α-shapes no cálculo da área e volume, observe os quatro discos superpostos da Fig. 14.

Fig. 14. Quatro discos superpostos (A). Complexo alpha representando as interseções (B).

Os discos são denotados por b_1, b_2, b_3 e b_4. A área do disco b_i é denotada por A_i, a área da interseção $b_i \cap b_j$ é A_{ij}, a área de $b_i \cap b_j \cap b_k$ é A_{ijk} e assim sucessivamente. Utilizando então o princípio da inclusão-exclusão, a fórmula resultante é:

$$A_{total} = (A_1 + A_2 + A_3 + A_4) - (A_{12} + A_{13} + A_{14} + A_{23} + A_{24} + A_{34}) + (A_{123} + A_{124} + A_{134} + A_{234}) - A_{1234}.$$

Esta fórmula contém seis elementos redundantes que se cancelam: $A_{13} = A_{123}, A_{14} = A_{124}$ e $A_{134} = A_{1234}$. É um desperdício realizar o cálculo utilizando esses termos redundantes. Se o complexo-α é construído, é possível guiar a fórmula de inclusão-exclusão, percorrendo um caminho que evita

redundâncias. Isto é feito da seguinte forma: adiciona-se a área de b_i se o vértice correspondente pertence ao complexo-α, subtrai-se a área de $b_i \cap b_j$ se a aresta correspondente existe no complexo-α e adiciona-se a área $b_i \cap b_j \cap b_k$ se o triângulo correspondente à interseção pertence ao complexo. Esta metodologia é chamada inclusão-exclusão direta. O resultado final é a fórmula que segue:

$$A_{total} = (A_1 + A_2 + A_3 + A_4) - (A_{12} + A_{23} + A_{24} + A_{34}) + A_{234}.$$

A fórmula da área total contém bem menos termos agora. Isto acontece porque a parte das regiões de Voronoi contidas em seu próprio átomo para os termos redundantes não se sobrepõem, ou seja, as interseções representadas pelos termos redundantes não correspondem a interseções das regiões de Voronoi. Observe também na equação reduzida que os termos contêm a interseção de no máximo três esferas, o que corresponde ao triângulo do complexo-α. No caso tridimensional a ordem das interseções é quatro, representada por um tetraedro no complexo-α.

Resumindo, a união de um conjunto de esferas pode ser expresso como um somatório de termos positivos e negativos, um para cada vértice, aresta, triângulo e tetraedro de seu complexo-α. Cada vértice corresponde a uma esfera, que deve se adicionar, cada aresta corresponde à interseção de duas esferas, a ser subtraída, os triângulos representam a interseção de três esferas, que são adicionadas e cada tetraedro corresponde à interseção de quatro esferas, que deve ser subtraída.

Para calcular a área e volume exatos de uma macromolécula, basta então construir o complexo-α desta molécula para α igual a zero. Assim tem-se uma representação da molécula com seus átomos em tamanho real. O segundo passo é aplicar o Algoritmo 1, de inclusão-exclusão direta, descrito a seguir.

```
for each σ ∩ C do
     if σ é um vértice i then
          V = V + vol(b_i);
          A = A + area(b_i);
      endif
      if σé uma aresta ij then
           V = V - vol(b_i ∩ b_j);
           A = A - area(b_i ∩ b_j);
      endif
      if σé um triângulo ijk then
           V = V + vol(b_i ∩ b_j ∩ b_k);
           A = A + area(b_i ∩ b_j ∩ b_k);
      endif
      if σé um tetraedro ijkl then
           V = V - vol(b_i ∩ b_j ∩ b_k ∩ b_l);
           A = A - area(b_i ∩ b_j ∩ b_k ∩ b_l);
      endif
endfor
```

Algoritmo 1. Regra da inclusão-exclusão direta.

Onde “V” é o volume total, “A” é a área total e “vol” e “area” são funções que calculam volume e área das interseções, respectivamente.

Observe que é possível calcular a área e o volume de moléculas de forma muito simples, se o complexo-α for construído. No entanto, fazer o cálculo das interseções entre três e principalmente quatro esferas é uma tarefa bastante complicada.

Os complexos-α podem ser utilizados também para identificar todas as cavidades existentes em uma proteína, além de medir sua área e volume. As cavidades (Liang et al., 1998b) são representadas como buracos da proteína inacessíveis a uma molécula de solvente. Utilizando o modelo acessível ao solvente (SA) as cavidades correspondem aos espaços vazios internos à macromolécula e que não possuem contato com o exterior. A Fig. 15 mostra uma molécula bidimensional em seus modelos Van der Waals e SA. Note as cavidades que surgem para o modelo SA da molécula.

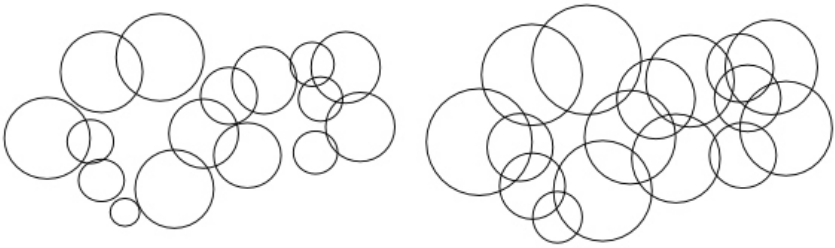


Fig. 15. Modelo VW da proteína (A) e modelo SA, definição de cavidades através do aumento dos raios dos átomos (B).

Os cálculos seguem a idéia de que para cada cavidade no modelo SA da proteína existe uma cavidade correspondente no seu complexo dual, e que a cavidade do modelo SA está contida na cavidade do complexo. A prova matemática para este fato é dada em (Edelsbrunner, 1995). Então, para identificar todas as cavidades da proteína primeiro são encontradas as cavidades do complexo dual. Após identificadas, sua área e volume são calculados com o auxílio da sua representação dual.

Uma cavidade no complexo dual é preenchida por tetraedros da triangulação de Delaunay, e que não pertencem ao complexo dual. Pode-se então representar a cavidade por este conjunto de tetraedros que a preenchem. Sua superfície será formada por um subconjunto dos triângulos que correspondem às faces desses tetraedros. Vários conjuntos de tetraedros servirão para representar todas as cavidades do complexo dual.

Para identificar os tetraedros que correspondem às cavidades, o procedimento é o seguinte: considere que o último simplex do complexo-α C definido para a molécula em seu modelo SA (isto é, com o raio de seus átomos igual o raio de van der Waals mais o raio da molécula de solvente) está numa posição J do filtro. O filtro deve então ser percorrido de trás para frente, a partir de sua última posição até a posição *J*+1, e todos os tetraedros do filtro devem ser avaliados. Aqueles que correspondem a cavidades no complexo-α *C* definido pela posição *J* do filtro devem ser identificados. Os tetraedros vão sendo alocados a conjuntos de tetraedros, cada um representando uma cavidade. Cada tetraedro pode pertencer a no máximo uma cavidade. O Algoritmo 2 explica com mais detalhes este procedimento.

```
for j := N down to J+1
if σj é um tetraedro then
     adicione σj a um novo conjunto vazio de tetraedros;
elseif σj é triângulo  then
      determine os dois tetraedros σ  e  σ' que compartilham σj;
      if σ  e  σ' pertencem a conjuntos de tetraedros diferentes
            then
                 faça a união dos conjuntos de σ  e σ'
                 transformando-os em apenas um conjunto;
            endif
      endif
endfor
```

Algoritmo 2. Procedimento para identificar os tetraedros que correspondem a cavidades.

Sendo *N* a última posição do filtro.

Para o funcionamento correto do algoritmo é necessário que para cada triângulo, os dois tetraedros que o compartilham já tenham sido adicionados ao sistema. A construção do filtro garante que isto seja sempre verdade.

O volume de uma cavidade no modelo SA pode ser calculado baseado na regra de inclusão-exclusão. Todas as fórmulas envolvidas no cálculo são provadas em (Edelsbrunner, 1995). O volume da cavidade representada pelo conjunto T de tetraedros fica:

$$V_{total} = V_0 - V_1 + V_2 - V_3$$

Os valores V_0, V_1, V_2 e V_3 correspondem ao somatório dos volumes de quatro tipos de objetos geométricos: tetraedros, setores de esferas, “wedges” de interseção de duas esferas e “halves” de interseções de três esferas.

$\mathbf{V_0}$ = Σ vol(σ), onde σ é um tetraedro. O somatório envolve todos os tetraedros de T.

$\mathbf{V_1}$ = Σ $\varphi_{v,\sigma}$× vol(b_v), onde σ ∈ T, ν ∈ *C* é um vértice de σ e b_v é uma esfera de centro *v*. vol(b_v) é o volume de b_v e $\varphi_{v,\sigma}$ é o ângulo sólido em v dentro de σ. O somatório ocorre para todo σ ∈ T e todo ν ∈ *C* de σ.

$\mathbf{V_2}$ = Σ $\varphi_{\varepsilon,\sigma}$× vol($b_v \cap b_\mu$), onde σ ∈ T, ε *C* é uma aresta de σ, e *v* e μ são os pontos extremos de ε. vol($b_v \cap b_\mu$) é o volume da interseção de duas bolas, e $\varphi_{\varepsilon,\sigma}$ é o ângulo diedral de ε dentro de σ.

$\mathbf{V_3}$ = Σ (1/2) × vol($b_v \cap b_\mu \cap b_\lambda$), onde *v*, μ e λ são vértices de um tetraedro σ ∈ *T* que possui um triângulo em *C*. vol($b_v \cap b_\mu \cap b_\lambda$) é o volume da interseção das três esferas.

Todos os ângulos são medidos em revolução, isto é, entre 0 e 1, onde 1 corresponde a 180 graus.

O mesmo método pode ser estendido para o cálculo da área. A fórmula da área total de uma cavidade representada por um conjunto de tetraedros T é:

$$\mathbf{A_{total}} = A_1 + A_2 + A_3$$

A_1, A_2 e A_3, assim como no cálculo do volume, correspondem às áreas dos setores de esferas, "wedges" das interseções de duas esferas e "halves" das interseções de três esferas.

$\mathbf{A_1}$ = Σ $\varphi_{v,\sigma}$× area(b_v). A notação é a mesma utilizada em V_1, sendo que area(b_v) é a área da esfera b_v.

$\mathbf{A_2}$ = Σ $\varphi_{\varepsilon,\sigma}$× area($b_v \cap b_\mu$), com notações iguais às de V_2, sendo que area($b_v \cap b_\mu$) corresponde à área da interseção de duas esferas.

$\mathbf{A^3}$ = Σ (1/2) × area($b_v \cap b_\mu \cap b_\lambda$), com notações iguais às de V_3, sendo que area($b_v \cap b_\mu \cap b_\lambda$) corresponde à área da interseção entre três esferas.

As fórmulas são então reduzidas ao cálculo de elementos primitivos. A forma como estes elementos primitivos são calculados pode ser encontrada em (Edelsbrunner & Fu, 1994).

Até o momento observou-se o que são os α-shapes e como podem ser utilizados para calcular parâmetros de macromoléculas. Descreve-se a seguir um algoritmo, originalmente proposto por Edelsbrunner & Mücke

(1994), para determinar os α-shapes, os complexos-α e o filtro de um conjunto de pontos ou átomos. Esta descrição requer a introdução de alguns conceitos e definições matemáticas.

Para definir matematicamente o conceito de α-shapes, deve-se inicialmente fazer algumas suposições. Considere um conjunto de pontos *S* num espaço qualquer $\Re^d$. Assuma que estes pontos estão em uma posição geral, para evitar casos especiais. Em terceira dimensão, isto quer dizer que não há quatro pontos em um mesmo plano nem cinco pontos em uma mesma esfera. Esta suposição garante que para todo $T \subseteq S$ com $|T| = k + 1 \leq d + 1$ (onde *d* é a dimensão do espaço de *S*), o politopo $\sigma_T = \text{conv}(T)$ (convex hull de *T*) tem exatamente dimensão *k*, logo é um *k*-simplex. Na prática, casos onde os pontos não estão em posição geral ocorrem, e isso pode levar a degenerações nos cálculos. Para evitar esses casos existem alguns artifícios como o Simulation of Simplicity (SOS) (Edelsbrunner & Mücke, 1990), que causa uma perturbação simbólica nos índices dos vértices, e a computação exata, baseada em tipos numéricos que utilizam representações exatas dos números, como o CORE::Expr da biblioteca CORE[7].

É preciso agora definir a noção de α-*exposto*. Considere uma determinada esfera *b* (ou círculo, no caso $\Re^2$) de raio α. Esta esfera é dita vazia se $b \cap S = \varnothing$. Assim, um *k*-simplex σ_T é dito α-exposto se existe uma esfera *b* com $T = \partial b \cap S$, sendo ∂b a superfície da esfera *b*. Esta idéia é ilustrada na Fig. 16, que mostra em duas dimensões um simplex α-exposto e um não-exposto.

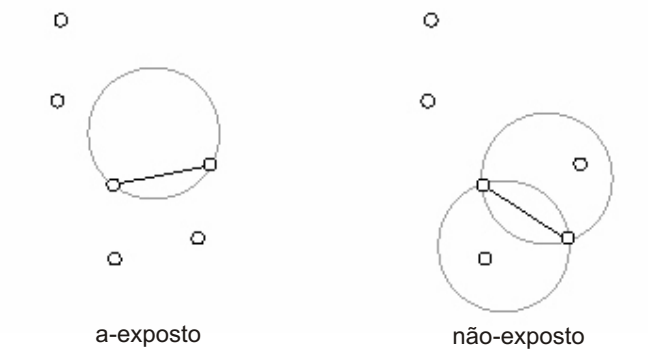


Fig. 16. Em duas dimensões, um simplex α-exposto e um não-exposto.

[7] www.cs.nyu.edu/exact/core/

A relação entre α-shape e a idéia de α-exposto é bastante simples. Como foi mencionado anteriormente, o α-shape para um dado α é um politopo determinado por todos os pontos de S nos quais uma esfera móvel de raio α consegue encostar. Isto é equivalente a dizer que os simplexos definidos por estes pontos são α-expostos.

Assim, a superfície ∂S_α de um α-shape de um conjunto de pontos S é o conjunto de todos os k-simplexos de S, para $0 \leq k < d$, que estão α-expostos,

$$\partial S_\alpha = \{\sigma_T \mid T \subseteq S, |T| \leq d \text{ e } \sigma_T \text{ é } \alpha\text{-exposto}\}.$$

O α-shape S_α de um conjunto de pontos S será então o politopo de superfície ∂S_α. Este politopo não é necessariamente convexo ou totalmente conexo.

Haverá sempre dois politopos definidos pela superfície ∂S_α, sendo que um deles corresponde ao espaço interno à superfície e o outro ao espaço externo. É claro que o α-shape corresponde ao politopo que não se estende ao infinito. Se as esferas definindo o α-shape são infinitesimalmente pequenas, tem-se que $S_\alpha = S$. Se o raio dessas esferas tendem ao infinito, tem-se que $S_\alpha = \text{conv}(S)$.

O espaço interno de um α-shape é composto de uma coleção de simplexos chamada complexo-α. Para construir o α-shape (ou mais precisamente, a superfície do α-shape) a melhor opção é primeiro determinar o complexo-α e depois analisar todos os seus simplexos. Aqueles que estão α-expostos pertencem ao α-shape.

Para definir formalmente o conceito de complexo-α, considere um k-simplex $\sigma_T = \text{conv}(T)$, tal que $0 \leq k \leq d$ e $T \subseteq S$. A esfera que circunscreve σ_T será denotada por b_T e o raio desta esfera por δ_T. Tem-se então que para um dado conjunto de pontos S no espaço $\Re^d$ e $0 \leq \alpha \leq \infty$, o complexo-$\alpha$ $C_\alpha(S)$ de S é um complexo simplicial da triangulação de Delaunay DT(S), tal que um simplex $\sigma_T \in \text{DT}(S)$ está em $C_\alpha(S)$ se:

(1) $\delta_T < \alpha$ e b_T é vazia, ou

(2) σ_T é uma face de outro simplex em $C_\alpha(S)$.

Ou seja, o complexo-α será formado pelo conjunto de todos os simplexos DT(S) que satisfazem a condição (1) mais todas as faces de DT(S) necessárias para transformar este conjunto em um complexo simplicial. A Fig. 17 faz uma comparação entre o α-shape, a triangulação de Delaunay e o complexo-α, construídos para um mesmo conjunto de pontos.

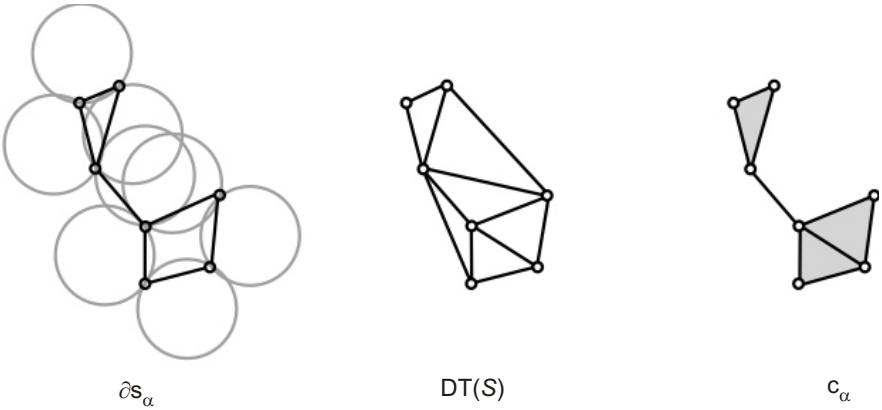


Fig. 17. Da esquerda para a direita, o α-shape, a triangulação de Delaunay e o complexo-α.

O α-shape é formado pela parte externa com complexo-α, e que ambos estão contidos na triangulação de Delaunay.

Agora que as idéias de α-shape e complexo-α foram melhor compreendidas, é possível elaborar um algoritmo para obter estas estruturas:

1. Calcule a triangulação de Delaunay de *S*.
2. Determine $C\alpha$ analisando todos os simplexos σ_T de DT(*S*). Se a esfera b_T, que circunscreve σ_T for vazia e o seu raio $\delta_T < \alpha$, então $\sigma_T \in C_\alpha$, juntamente com todas as suas faces.
3. Todos os *d*-simplexos de C_α compõem o interior de S_α. Todos os simplexos da superfície de C_α formam ∂S_α.

Este algoritmo, originalmente proposto por Edelsbrunner e Mücke (1994), apresenta duas vantagens. A primeira é que o algoritmo não é executado para cada valor de α; ao invés disso, ele computa uma representação implícita que pode ser utilizada para determinar o ∂S_α de qualquer α. Para cada simplex σ_T de DT(*S*) existe um intervalo único em que σ_T é uma face de S_α se e apenas se α está contido neste intervalo. O que o algoritmo faz é simplesmente associar um intervalo a cada simplex da triangulação.

A Segunda vantagem é que ele faz uma distinção entre os simplexos de DT(*S*), classificando-os em três tipos:

- **Interior** - se σ_T não pertence a ∂S_α, isto é, não está α-exposto;
- **Singular** - se σ_T pertence a ∂S_α e é face de um simplex de dimensão maior;
- **Regular** - se σ_T pertence a ∂S_α e não é face de nenhum simplex de dimensão maior.

Note que há arestas e triângulos de Delaunay que nunca podem ser singulares, caso a esfera b_T que os circunscreve englobe outros pontos de S. É conveniente então classificar um simplex σ_T de:

- **Atachado** - se $|T| = 2,3$ e $b_T \cap S \neq \emptyset$, e
- **Não-atachado** - caso contrário.

O intervalo para o qual um simplex pertence a C_α pode ser subdividido em partes, se forem introduzidos os valores $\underline{\mu}_T$ e $\overline{\mu}_T$, que representam o momento em que um simplex passa de interior para singular e de singular para regular, respectivamente. Se σ_T é um tetraedro, então $\underline{\mu}_T = \overline{\mu}_T = \delta_T$. Para os outros simplexos, $\underline{\mu}_T$ corresponde ao momento em que surge no complexo-α o primeiro simplex σ_T da triangulação de Delaunay que tem dimensão imediatamente maior, isto é, com $k' = k + 1$ (ou $|T'| = |T| + 1$), e cujo σ_T é face. $\underline{\mu}_T$ é o menor valor de α para o qual todos os d-simplexos dos quais σ_T é face pertencem completamente ao complexo.

De maneira mais formal, se considerar que $up_1(\alpha_T)$ é o conjunto de simplexos α_T, tal que $|T'| = |T| + 1$, pode-*se* definir:

- $\underline{\mu}_T$ =*min*($\{\delta_T | \sigma_T \in up_1(\sigma_T)$, *não-atachado*}"unido a" $\{\underline{\mu}_{T'} | \sigma_{T'} \in up_1(\sigma_T)$, *atachado*}), e
- $\overline{\mu}_T = \max(\{\overline{\mu}_{T'} | \sigma_{T'} \in up_1(\sigma_T)\})$.

Se o simplex em questão é uma aresta, todos os triângulos que contêm o simplex serão avaliados. $\underline{\mu}_T$ será o menor valor entre os raios δ dos triângulos não-atachados, e os limiares $\underline{\mu}_T$ dos atachados. O $\overline{\mu}_T$ da aresta será o máximo limiar $\overline{\mu}_T$ entre todos os triângulos, o que corresponde a dizer que todos os tetraedros incidentes a todos os triângulos (que são os mesmos tetraedros incidentes à aresta) foram definidos no complexo-α.

Como resultado, é possível especificar todos os intervalos de valores de α para os quais um simplex qualquer σ_T pertence a C_α. Os intervalos para vértices, arestas, triângulos e tetraedros são mostrados na Tabela 1.

Tabela 1. Intervalos e classificação de todos os simplexos de uma triangulação.

σ_T é...		*Singular*	*Regular*	*Interior*
Tetraedro				$(\delta_T, \infty]$
Aresta ou triângulo,	$\notin \partial conv(S)$, *não – atachado*	$(\delta_T, \underline{\mu}_T)$	$(\underline{\mu}_T, \overline{\mu}_T)$	$(\overline{\mu}_T, \infty]$
	$\notin \partial conv(S)$, *atachado*		$(\underline{\mu}_T, \overline{\mu}_T)$	$(\overline{\mu}_T, \infty]$
	$\in \partial conv(S)$, *não – atachado*	$(\delta_T, \underline{\mu}_T)$	$(\underline{\mu}_T, \infty]$	
	$\in \partial conv(S)$, *atachado*		$(\underline{\mu}_T, \infty]$	
Vértice,	$\notin \partial conv(S)$	$[0, \underline{\mu}_T)$	$(\underline{\mu}_T, \overline{\mu}_T)$	$(\overline{\mu}_T, \infty]$
	$\in \partial conv(S)$	$[0, \underline{\mu}_T)$	$(\underline{\mu}_T, \infty]$	

Para armazenar os intervalos de forma adequada no computador, a representação adotada é uma estrutura com três valores (δ_T, $\underline{\mu}_T$, $\overline{\mu}_T$), denotada pela letra Δ. Note que nos casos em que o simplex é atachado, $\Delta(\delta_T)$ será indefinido, e assumirá valor -2 na estrutura. Nos casos em que o simplex pertence ao convex hull ele nunca será interior, portanto $\Delta(\overline{\mu}_T)$ será infinito, e assumirá valor -1 na estrutura.

Para entender melhor os valores mostrados na Tabela 1, observe a seguinte explicação, dada em (Mücke, 1993). Considere um triângulo $\sigma_T \in$ DT(S), com $T = \{p_i, p_j, p_k\}$, que não pertence ao convex hull de S. Considere também dois tetraedros $\sigma_{T'}$ e $\sigma_{T''}$ incidentes a σ_T e pertencentes a DT(S), e assuma $T' = T$ "unido a" $\{p_u\}$e $T'' = T$ "unido a" $\{p_v\}$. Além disso, $0 < \delta_T < \delta_{T''} < \infty$ ou seja, $\underline{\mu}_T = \delta_{T'}$ e $\overline{\mu}_T = \delta_{T''}$. Agora, fixe um valor de α. Se $\delta_{T''} < \alpha \leq \infty$, então o triângulo não é α-exposto. Entretanto, ele será parte do interior de S_α, porque ambos os tetraedros incidentes estão em C_α. Se $\delta_{T'} < \alpha < \delta_{T''}$, então o triângulo é α-exposto e $\sigma_{T'}$ está em C_α mas $\sigma_{T'}$ não está. Isto significa que $\sigma_{T'}$ é um triângulo regular de C_α. Para $\alpha < \delta_T$, nem $\sigma_{T'}$ ou $\sigma_{T''}$ são tetraedros de S_α, mas σ_T pode ainda ser um triângulo singular, se e apenas se $\delta_T < \alpha$, e nem p_u ou p_v estão dentro de b_T. Se um dos dois pontos está dentro de b_T, então σ_T é atachado, e σ_T nunca será um triângulo singular de C_α, não importa qual o valor de α.

O complexo-α será formado por todos os simplexos interiores, regulares e singulares para um dado valor de α. O interior do α-shape é triangulado pelos simplexos interiores. A parte externa do complexo será formada pelo conjunto de triângulos regulares e suas arestas e vértices.

Os limiares dos intervalos da Tabela 1 são chamados de limiares-α. Já que todo $\underline{\mu}_T$ e $\overline{\mu}_T$ são valores de δ de outros simplexos (isto é, são valores dos raios das esferas que circunscrevem outros simplexos), cada limiar-α é o raio de um simplex em DT(S). Ou seja, o conjunto de limiares-α é o conjunto

de raios de todos os *k*-simplexos não-atachados para $1 \leq k < 3$. A seqüência ordenada de limiares-α será chamada espectro-α.

Como dito anteriormente, o algoritmo calcula o complexo-α através de uma representação implícita, que corresponde aos intervalos de cada simplex. O objetivo então é calcular os limiares-α de todos os simplexos da triangulação de Delaunay. Com estes valores é possível obter o complexo-α e o α-shape para qualquer valor de α. Para um *d*-simplex (um tetraedro) este cálculo é bastante simples. A esfera que o circunscreve é vazia por definição, isto é, todo tetraedro da triangulação de Delaunay é não-atachado. Assim, um tetraedro σ_T pertence ao complexo-α apenas se $\delta_T < \alpha$. Como um *d*-simplex será sempre interior a C_α, então $\underline{\mu}_T = \overline{\mu}_T = \delta_T$, para todo tetraedro.

Após calcular os intervalos de todos os *d*-simplexos, é possível prosseguir com o cálculo para os *k*-simplexos, tal que $k < d$. A idéia aqui é computar primeiro os intervalos dos simplexos de maior dimensão e utilizar esses valores para computar os intervalos dos de menor dimensão, como ficará mais claro a seguir. Considere um *k*-simplex σ_T, com $k < d$. Se σ_T for não-atachado, ele pertencerá ao complexo-α se $\delta_T < \alpha$ então $\Delta(\delta_T)$ será igual a δ_T. Caso contrário, a única maneira de σ_T pertencer a C_α é se um dos simplexos $\sigma_{T'}$ de dimensão maior que incidem sobre ele ($\sigma_{T'} \in up_1(\sigma_T)$) também pertença a C_α. Neste caso, $\Delta(\delta_T) = -2$.

Os valores de $\underline{\mu}_T$ para todo *k*-simplex podem ser facilmente determinados utilizando os intervalos já calculados para os (*k*+1)-simplexos. O procedimento para o cálculo de $\underline{\mu}_T$ pode ser descrito de forma algorítmica (Agoritmo 3) a seguir.

```
for k := d-1 down to 2 do
   for each k-simplex σ_T of DT(S) do
       μ_T := ∞;
       for each (k+1)-simplex σ_T' ∈ up_1(σ_T) do
          if σ_T' não-atachado then
                tmp := δ_T';
          else
          tmp := μ_T';
          endif
          μ_T := min(tmp, μ_T);
       endfor
   endfor
endfor
```

Algoritmo 3. Cálculo dos limiares $\underline{\mu}_T$.

A mesma idéia pode ser aplicada para o cálculo de $\overline{\mu}_T$ (Algoritmo 4).

```
for k := d-1 down to 2 do
    for each k-simplex σ_T of DT(S) do
        μ̄_T := 0;
        if σ_T ∈ conv(S) then
            μ̄_T := ∞;
        else
            for each (k+1)-simplex σ_T' ∈ up_1(σ_T) do
                tmp := μ̄_T;
                μ̄_T := max(tmp, μ̄_T);
            endfor
        endif
    endfor
endfor
```

Algoritmo 4. Cálculo dos limiares $\bar{\mu}_T$.

Todos os valores de Δ podem ser calculados no mesmo loop. Agora é possível escrever um algoritmo genérico que determina os intervalos de todos os simplexos de uma triangulação de Delaunay (Algoritmo 5).

```
for k := d down to 2 do
    for each k-simplex σ_T of DT(S) do
        if k = d then //se σ_T é um tetraedro
            μ̄_T := δ_T;
            μ̲_T := δ_T;

        else //triângulos, arestas e vértices
            μ̲_T := ∞;
            μ̄_T := 0;

            for each (k+1)-simplex σ_T' ∈ up_1(σ_T) do
                max_alpha := μ̄_T';

                if σ_T' não-atachado then
                    min_alpha := δ_T';

                Else
                min_alpha := μ̲_T' ;

                endif
                μ̲_T := min(min_alpha, μ̲_T);
                μ̄_T := max(max_alpha, μ̄_T);
            Endfor

            if σ_T ∈ conv(S) then
                μ̄_T := ∞;
            endif
        endif
    endfor
Endfor
```

Algoritmo 5. Procedimento para determinar os invervalos de todos os simplexos.

Para implementar este algoritmo é necessário ter a triangulação de Delaunay já construída. Algoritmos para a construção da triangulação podem ser facilmente encontrados na literatura (Sugihara, 2007).

Após computados os intervalos, se torna trivial determinar quais simplexos compõem o complexo-α e o α-shape. Se fixado um valor de α, pertencerão ao complexo todos os simplexos que já foram definidos até este valor, ou seja, aqueles não-atachados que possuem $\Delta(\delta_\tau) \leq \alpha$ e aqueles atachados com $\Delta(\mu_\tau) \leq \alpha$. Os vértices sempre estarão no complexo para qualquer valor positivo de α. Para saber quem pertence à superfície do α-shape, basta selecionar todos os simplexos regulares e singulares, ou seja, com $\Delta(\mu_\tau) \leq \alpha$, excluindo os tetraedros.

Quando se trata de alpha shapes ponderados, o algoritmo para o cálculo dos intervalos é o mesmo. A diferença é que se deve utilizar uma triangulação regular, ao invés da triangulação de Delaunay. Isto, obviamente traz diversas implicações no cálculo dos intervalos, pois todas as primitivas e predicados necessários devem levar em consideração a ponderação.

Para realizar as implementações, utiliza-se uma biblioteca de geometria computacional disponível gratuitamente na internet chamada CGAL. Esta biblioteca é escrita em linguagem C++ e o seu *download* pode ser feito em http://www.cgal.org.

A CGAL é uma biblioteca desenvolvida por um conjunto de sete instituições: Utrecht University (Holanda), ETH Zurich (Suiça), Free University Berlin (Alemanha), INRIA Sophia-Antipolis (França), Max Planck Institute Saarbrucken (Alemanha), RISC Linz (Áustria) e Tell Aviv University (Israel). O objetivo destas entidades é disponibilizar para a comunidade científica implementações capazes de contornar problemas comuns envolvidos em algoritmos de geometria computacional, como degenerações causadas por cálculos inexatos, falta de generalidade dos algoritmos e complexidade inerente a soluções eficientes.

A biblioteca é estruturada em três blocos principais de algoritmos. O primeiro deles, chamado *kernel*, consiste de primitivas, objetos geométricos de tamanho constante (pontos, esferas, linhas, etc.) e predicados para estes objetos (testes de orientação de pontos, testes de interseção, etc.). A segunda parte contém uma série de algoritmos e estruturas de dados geométricos padrões, como convex hull e triangulações. A última parte da CGAL consiste de uma biblioteca de suporte para I/O, visualização, geradores aleatórios, e outros.

A CGAL possui suporte a uma série de representações numéricas, incluindo representações exatas. Isto dá ao usuário a opção de realizar cálculos exatos, sem erros de arredondamento, ou, se para uma dada aplicação a velocidade for mais importante que a precisão, é possível mudar para uma representação *float* ou *double*. Como sua estrutura é toda baseada em *templates*, pode-se realizar este tipo de mudança sem a necessidade de reescrever o programa. Estas características conferem propriedades muito importantes à CGAL, como robustez e generalidade.

Além da vantagem de ser gratuita e oferecer suporte para computações exatas, a CGAL possui implementações de algoritmos para a construção da Triangulação de Delaunay, Triangulação Regular e Alpha Shapes (ponderado ou não), o que a torna perfeitamente adequada ao nosso trabalho. A contribuição da CGAL foi muito importante para a realização deste projeto, pois a implementação desses algoritmos é muito complexa, principalmente o da triangulação regular.

Muitas proteínas são compostas por uma combinação de cadeias protéicas que se encaixam uma na outra em posições específicas, baseadas na complementaridade de seu formato tridimensional e em interações físico-químicas de seus resíduos, como forças eletrostáticas, hidrofílicas e hidrofóbicas, etc. Estudar como se dá a interação entre as cadeias, de forma a determinar que propriedades específicas de suas superfícies levam ao encaixe, pode ajudar a entender vários processos envolvidos na conformação tridimensional das proteínas, a classificar tipos de interação, predizer probabilidades de acoplamento e projetar moléculas que interajam com proteínas de interesse.

Para estudar essas interações, a modelagem e visualização da interface de contato entre as proteínas pode ser um artifício muito útil. Além de prover ao especialista uma noção intuitiva do encaixe tridimensional, é possível realizar cálculos de parâmetros estruturais envolvendo esta interface (como volume e área) e criar mapas de interações que permitem que comparações entre as proteínas sejam feitas em duas dimensões.

Descreve-se a seguir um algoritmo desenvolvido para a visualização da interface entre duas cadeias de proteínas que se baseia na representação ponderada de Delaunay das macromoléculas.

Se uma proteína é totalmente decomposta em tetraedros, o espaço que corresponde à interface entre suas cadeias também será totalmente triangulado, sendo preenchido por tetraedros contendo átomos das duas cadeias. A Fig. 18 ilustra em duas dimensões o que ocorre na triangulação da interface.

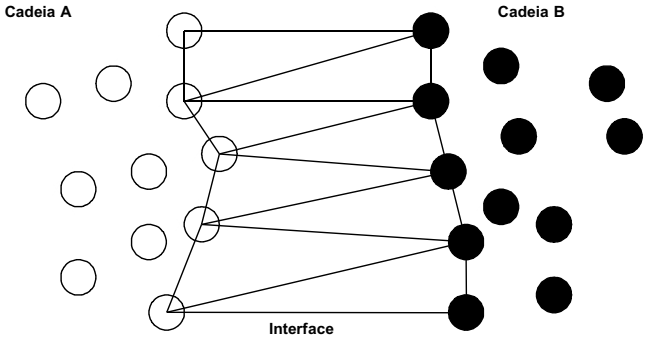


Fig. 18. Interface entre duas cadeias protéicas, destaque para a triangulação da interface.

O espaço entre as duas cadeias é completamente preenchido por triângulos. Entretanto, para moléculas reais, em três dimensões, os triângulos destacados na Fig. 18 são na verdade tetraedros. Ainda observando a Fig. 18, nota-se que há algumas maneiras bem simples de modelar a superfície de contato. Uma delas é conectar os centros das arestas que interligam as duas cadeias. O resultado é um superfície única de contato. Outra maneira consiste em eliminar estas arestas, gerando duas superfícies de contato, uma para cada cadeia. Ambos os casos são mostrados na Fig. 19.

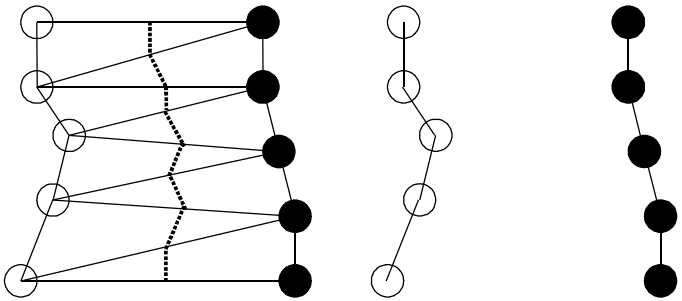


Fig. 19. Duas maneiras simples de obter a superfície de contato.

Estas opções, embora simples de serem aplicadas, não levam em consideração o raio de van der Waals dos átomos, e portanto, não correspondem à superfície de contato real entre as cadeias. Uma abordagem mais interessante é modelar a superfície utilizando os pontos de cruzamento entre a superfície de cada átomo (representado como uma esfera com o raio de van der Waals) e as arestas. Esses pontos são então interligados, gerando duas novas superfícies, assim como mostrado de forma simplificada na Fig. 20.

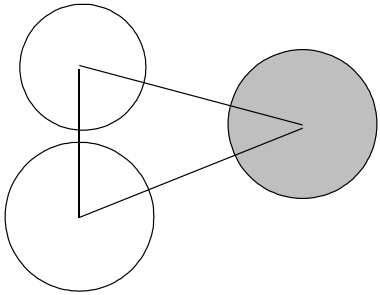


Fig. 20. Obtendo a superfície real, considerando agora o raio de van der Waals dos átomos.

A única dificuldade desta representação é que a nova superfície é formada por pontos que não pertencem aos dados originais, isto é, à triangulação regular. Entretanto, esses pontos podem ser facilmente calculados sabendo-se o raio de van der Waals de cada átomo, como mostrado na Fig. 21. Outro detalhe é que em três dimensões os cálculos devem ser feitos sobre tetraedros, e não triângulos.

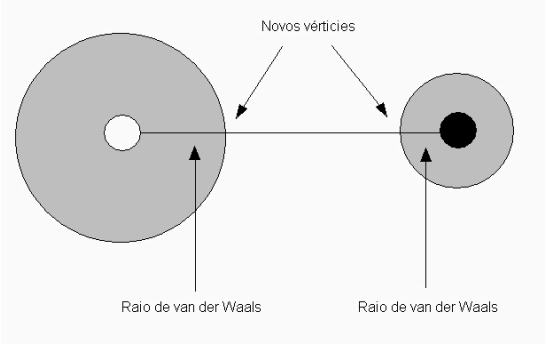


Fig. 21. Vértices que vão compor a superfície de contato, considerando o raio de van der Waals.

Na interface, dois tipos de tetraedros são possíveis. Aqueles que contêm três átomos de uma cadeia e um da outra e aqueles com dois átomos de cada cadeia. O tipo de tetraedro determina como ele será dividido para dar origem às superfícies. A Fig. 22 mostra o que acontece em cada caso.

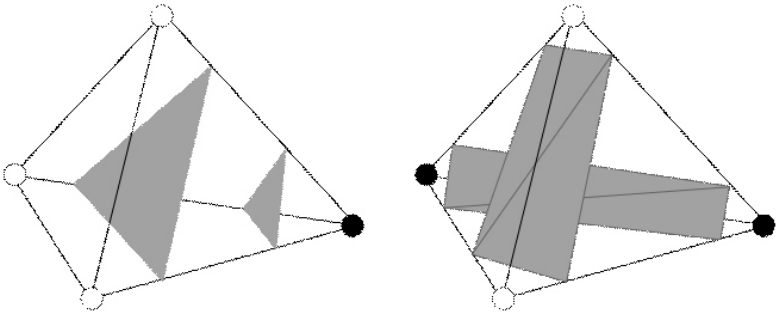


Fig. 22. Decomposição dos tetraedros em superfícies, para os dois casos específicos - três átomos de uma cadeia e um de outra e dois átomos de cada cadeia.

É claro que os vértices que definem as superfícies são determinados pelo raio de van der Waals. É possível também utilizar a metade da distância entre os novos vértices como o ponto que define a superfície, assim como feito em Ray et al. (2003). O resultado seria uma única superfície representando a interface. Esta abordagem, no entanto, apresenta desvantagens. Se duas superfícies são utilizadas, é possível analisar os resíduos em cada uma das cadeias separadamente, estudando os diversos tipos de interações locais entre os aminoácidos. Com apenas uma superfície, toda a informação proveniente deste tipo de interação local é perdida.

De forma resumida, o algoritmo para a construção das superfícies de contato entre duas cadeias pode ser descrito da seguinte forma:

1. construa a triangulação regular da molécula;
2. procure na sua triangulação todos os tetraedros que contêm átomos das duas cadeias e armazene-os numa lista;
3. para cada tetraedro, identifique o tipo (número de átomos de cada cadeia) e determine os novos vértices;
4. desenhar as superfícies.

As superfícies (Fig. 23 e 24) calculadas foram desenhadas com a ajuda do aplicativo JavaView[8]. Mostra-se a seguir, as superfícies desenhadas de algumas proteínas, calculadas utilizando a implementação proposta.

[8] http://www.zib.de/javaview/

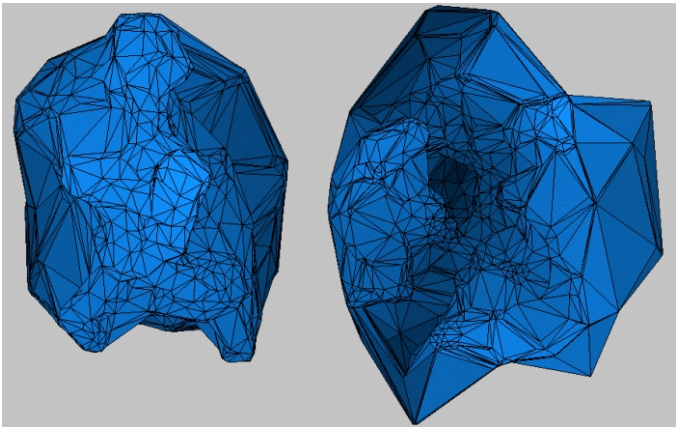


Fig. 23. Superfícies de contato obtidas para as duas cadeias da proteína 1CHO.

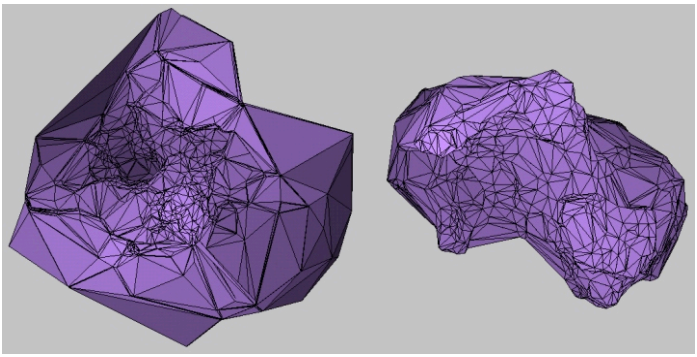


Fig. 24. Superfícies de contato obtidas para as duas cadeias da proteína 1FSS.

Observe como as formas são complementares. Além de calcular a superfície das proteínas, o programa também associa cada vértice ao seu átomo e resíduo. Assim, é possível implementar mapas bidimensionais com triângulos coloridos de acordo com alguma propriedade físico-química que se deseja estudar, como potencial eletrostático, hidrofobicidade, curvatura etc., e assim analisar interações locais entre as cadeias. Todos esses parâmetros estão disponíveis no JPD (Neshich et al., 2004).

Resultados e Discussão

A implementação de um algoritmo para calcular volume e área das proteínas e de suas cavidades utilizando complexos-α pode ser dividida em duas etapas principais. A primeira consiste no complexo-α em si. É preciso implementar um programa que receba como entrada um arquivo do PDB[9], extraia dele todos os átomos, suas coordenadas e seus raios, construa a triangulação regular desses dados e depois determine os intervalos-α de todos os simplexos. A segunda parte é utilizar os complexos-α para calcular os parâmetros. Isto implica que além do algoritmo que calcula os complexos-α ter de prover funções que retornem todos os simplexos do complexo para um determinado α, é necessário implementar funções para o cálculo do volume e área da interseção de duas, três e quatro esferas.

Para a leitura dos arquivos PDB foi utilizada uma biblioteca já existente, pertencente ao STING (Neshich et al., 2003), que retira as coordenadas dos átomos, seus resíduos e cadeias, armazenando essas informações em estruturas especiais. Os raios foram obtidos em um arquivo padrão contendo os raios de van der Waals de todos os átomos para todos os resíduos. Para a construção dos complexos-α, foram utilizadas as bibliotecas da CGAL. No entanto, foi necessário implementar códigos para possibilitar a integração das duas interfaces, isto é, a leitura do PDB e a construção do complexo.

Infelizmente, a classe *Alpha_shape_3*, provida pela CGAL, responsável pela construção dos complexos-α não possui todas as funcionalidades necessárias para o cálculo dos parâmetros. Primeiro, ela não calcula os intervalos para as arestas, o faz apenas para triângulos e tetraedros, talvez porque a complexidade do cálculo é bem maior no caso das arestas. Segundo, ela não possui funções que retornam os simplexos pertencentes ao complexo, apenas calcula para um dado valor de α o número de simplexos regulares, singulares e interiores, e o seu tipo, isto é, triângulo ou tetraedro.

Este problema foi resolvido utilizando hierarquia de classes. Foi implementada uma classe descendente da *Alpha_shape_3* que aproveita todas as suas funcionalidades, mas que calcula o intervalo das arestas e que também gera uma lista com todos os simplexos presentes no complexo-α. A hierarquia representa uma excelente solução, pois não é necessário alterar as classes originais da CGAL.

[9] http://www.pdb.org

Foram também encontrados alguns erros na biblioteca. Esses erros foram notificados aos responsáveis pelo desenvolvimento da CGAL, com as devidas sugestões para correção. Os erros apontados foram corrigidos na última versão (CGAL-3.0.1 - *bug-fix release*), e alguns deles são inclusive destacados na página, como a correção do método para classificação de arestas (classe *Alpha_shape_3*).

A etapa final consiste em implementar funções para o cálculo do volume e área das interseções de duas, três e quatro esferas. Infelizmente, fórmulas explícitas para estes cálculos só existem para esferas que possuem raios iguais (Powell, 1964; Helte, 1994). Foi encontrada apenas uma referência na literatura que aborda este assunto (Edelsbrunner & Fu, 1994). A metodologia utilizada é bastante complexa e consiste no cálculo de uma série de "blocos" mais simples, como discos, setores de esferas, ângulos diedrais, etc., que são combinados para compor os valores finais das interseções. O cálculo envolve um conjunto de mais de 100 funções.

Para avaliar a eficiência da implementação, foram feitas algumas comparações com *softwares* disponíveis gratuitamente na internet, como o Pocket[10] e o AlphaShapes[11] versão 4.1. A Tabela 2 mostra os resultados obtidos para algumas proteínas.

Tabela 2. Comparação dos resultados obtidos para algumas proteínas.

				INTER		*Pocket*		*Alpha Shapes-4.1*	
Proteína	*Átomo*	*Cadeia*	*Parâmetro*	*VW*	*SA*	*VW*	*SA*	*VW*	*SA*
16vp	2457	1	Volume	31157.7	59457.7	?	59459.9	31157.8	59459.9
			Area	31997.4	14230.3	?	14243.3	32001.8	14243.3
14gs	3067	2	Volume	38822.9	73324.8	?	73341.1	38823.2	73346.3
			Area	39783	16188.6	?	16292.1	39797.9	16289.8
1ar9	6653	5	Volume	83411.4	157626	?	157595	83416.7	157630
			Area	85892.8	34884.2	?	34900.1	85886.2	34906.2
1a0t	9922	3	Volume	119469	220103	?	220174	119469	220176
			Area	122699	42883.9	?	43138.5	122705	43103.9
1bgy	30507	22	Volume	390703	752118	?	?	390711	752180
			Area	385825	165949	?	?	385967	166250

A implementação descrita neste trabalho é chamada INTER. Observe na Tabela 2 que os resultados se aproximam muito, mas há uma semelhança maior entre os resultados do Pocket e do AlphaShapes. A discrepância observada é resultado de imprecisões relativas a aproximações numéricas.
No algoritmo INTER Foi utilizado o tipo *double*, enquanto o Pocket utiliza representação exata e o AlphaShapes utiliza a técnica de *Simulation of Simplicity* para evitar degenerações e realizar computação exata. Inicial o algoritmo INTER também utilizava representação exata, através da biblioteca CORE, entretanto, foi encontrado um *erro* no tipo numérico

[10] http://biogeometry.duke.edu/software/proshape/index.html
[11] http://www.alphashapes.org

CORE::Expr que atrasa muito a execução do código, tornando-a inviável. Mesmo assim, os resultados usando *double* foram considerados muito bons, pois não comprometem as avaliações biológicas envolvendo volume e área.

É interessante observar, porém, que mesmo utilizando representação inexata, em algumas situações os resultados do algoritmo INTER parecem mais consistentes. É o caso da proteína 1ar9, em que o volume SA calculado pelo AlphaShapes se aproxima mais dos resultados do INTER que do Pocket.

Além disso, veja que o Pocket não é capaz de realizar os cálculos para o modelo de VW, e se as proteínas são muito grandes, o programa simplesmente falha, como para a 1bgy. O AlphaShapes calcula o modelo VW, mas se a proteína for muito grande (maior que a 1bgy) o programa também não consegue ser executado. Isso acontece, por exemplo, para a proteína 1gav, que possui mais de 40.000 átomos.

Conclusões

O universo das proteínas é regido por conjuntos, ou módulos, de interações moleculares que não são bem compreendidas. Esses eventos ocorrem constantemente em nossas células a velocidades assustadoras, assumindo a estrutura de redes complexas de interações intra e inter-modulares. Os biólogos e bioquímicos precisam de pistas sobre a funcionalidade das proteínas, como elas interagem, onde elas se localizam, etc., a fim de compreender esse universo misterioso. Técnicas experimentais modernas têm produzido grandes volumes de informação sobre essas importantes moléculas, e cabe à bioinformática extrair conhecimento relevante dessa enorme massa de dados, tentando assim responder às perguntas mais pertinentes dos biólogos.

Para isso, é necessário aplicar conhecimentos matemáticos avançados aos problemas da biologia. Os dados biológicos mais recentes, referentes a padrões de interações moleculares, expressão gênica e estruturas de proteínas, são todos numéricos e merecem um tratamento estatístico e matemático bastante rigoroso. Essa interação de áreas de tradições distintas como a matemática e a biologia já é uma realidade em todo mundo e tem produzido resultados extremamente promissores para a ciência. Através de implementações computacionais de algoritmos matemáticos, os cientistas entendem cada vez melhor como funcionam os complexos sistemas biológicos e hoje já é possível projetar moléculas que interajam com proteínas de interesse e com isso produzir novos medicamentos, avaliar quais os genes e que produtos do DNA são responsáveis por determinados tipos de câncer e entender doenças relacionadas ao mal funcionamento de proteínas causados por defeitos genéticos.

Neste trabalho foram apresentados alguns conceitos matemáticos extremamente importantes para a biologia molecular. Embora tenham sido originalmente propostos para outros fins, essas idéias encontraram nos dados biológicos uma variedade de aplicações, e têm sido empregadas com sucesso a problemas relacionados à estrutura das proteínas. Foi demonstrado que através da triangulação Regular de uma macromolécula é possível derivar um algoritmo capaz de extrair e modelar a interface de contato entre duas cadeias protéicas. Este algoritmo foi implementado e testado, e os resultados obtidos foram bastante satisfatórios. Em um trabalho futuro, será desenvolvido um algoritmo que gere mapas bidimensionais através da projeção destas superfícies em um plano. Os mapas serão coloridos para destacar a presença de cada resíduo e assim melhor estudar as interações locais presentes na interface. O objetivo é desenvolver um *software* a ser disponibilizado na internet.

O conceito de α-*shapes* como uma técnica que possui grande potencial para o cálculo de parâmetros estruturais de proteínas foi bastante explorado aqui. Foi desenvolvido um algoritmo que realiza o cálculo do volume e superfície dessas moléculas de forma eficiente. A próxima etapa é realizar o mesmo cálculo para as cavidades. Além disso, será implementado uma forma de discriminar a contribuição individual de cada átomo na superfície e volume total das proteínas.

Referências

ALBERT, B.; JOHNSON, A.; LEWIS, J.; RAFF, M.; ROBERTS, K.; WALTER, K. *Molecular biology of the cell*. 4. ed. Mew York: Garland Science, 2002.

ALDEN, C.; KIM, S.H. Molecular biology of the cell. *J. Mol. Biol.*, v. 132, p. 411-434, 1979.

AURENHAMMER, F. Voronoi diagrams a survey of a fundamental geometric data structure. *ACM Computing Surveys*, v. 23, n. 5, p. 345-405, Sept. 1991.

BERMAN, H. M.; WESTBROOK, J.; FENG, Z.; GILLILAND, G.; BHAT, T. N.; WEISSIG, H.; SHINDYALOV, I. N.; BOURNE, P. E. The Protein Data Bank. *Nucl. Acid Res.,* v. 28, n. 1, p. 235-242, Jan. 2000.

BRANDEN, C.; TOOZE, J. *Introduction to protein structure*. 2nd ed. New York: Garland Science, 1999.

CHAZELLE, B.; DEVILLERS, O.; HURTADO, F.; MORA, M.; SACRISTÁN, V.; TEILLAUD, M. *Splitting a Delaunay triangulation in linear time*. Sophia Antipolis: INRIA, 2001. 12 p. (Rapport de recherché, 4160). Disponível em: <ftp://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-4160.pdf>. Acesso em: 10 jun. 2007.

CONNOLLY, M. L. Analytical molecular surface calculation. *J. Appl. Cryst.*, v. 16, p. 548-558, 1983.

CONNOLLY, M. L. Computation of molecular volume. *J. Am. Chem. Soc.*, v. 107, p. 1118-1124, 1985a.

CONNOLLY, M. L. Molecular surface triangulation. *J. Appl. Cryst.*, v. 18, p. 499-505, 1985b.

DAVID, E.; DAVID, C. Voronoi polyhedra as a tool for studying salvation structure. *J. Chem. Phys.*, v. 76, p. 4611-4614, 1982.

DELAUNAY, B. Sur la sphère vide. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, v. 7, p. 793-800, 1934.

EDELSBRUNNER, H. The union of balls and its dual shape. *Discrete and Computational Geometry*, v. 13, p. 415-440, 1985.

EDELSBRUNNER, H. *Weighted alpha shapes*. Champaign, IL: University of Illinois at Urbana-Champaign - Dept. Comput. Sci., 1992. (Technical Report UIUCDCSR, 921760).

EDELSBRUNNER, H.; FU, P. *Measuring space filling diagrams and voids*. Champaign, IL: University of Illinois at Urbana-Champaign: Beckman Inst. - Molecular Biophysics Group, 1994. (Report UIUC-BI-MB-94-01).

EDELSBRUNNER, H.; MÜCKE, E. Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics*, v. 9, n. 1, p. 66-104, Jan. 1990.

EDELSBRUNNER, H.; MÜCKE, E. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, v. 13, n. 1, p. 43-72, Jan. 1994.

NESHICH, G.; MANCINI, A.; BAUDET, C.; YAMAGISHI, M. E. B.; FALCÃO, P. R. K.; FILETO, R.; ROCCHIA, W.; PINTO, I.; MONTAGNER, A.; PALANDRANI, J.; KRAUCHENKO, J.; TORRES, R. C.; SOUZA, S.; TOGAWA, R. C.; HIGA, R. H.; Java Protein Dossier: a novel Web based data visualization tool for comprehensive analysis of protein structure. *Nucleic Acid Research***,** v. 32, p. W595-W601, 2004.

NESHICH, G.; TOGAWA, R. C.; MANCINI, A. L.; KUSER, P. R.; YAMAGISHI, M. E. B.; PAPPAS JUNIOR, G.; TORRES, W. V.; CAMPOS, T. F. e; FERREIRA, L. L.; LUNA, F. M.; OLIVEIRA, A. G.; MIURA, R. T.; INOUE, M. K.; HORITA, L. G.; SOUZA, D. F. de; DOMINIQUINI, F.; ÁLVARO, A.; LIMA, C. S.; OGAWA, F. O.; GOMES, G. B.; PALANDRANI, J. F.; SANTOS, G. F. dos; FREITAS, E. M. de; MATTIUZ, A. R..; COSTA, I. C.; ALMEIDA, C. L. de; SOUZA, S.; BAUDET, C.; HIGA, R. H. STING Millennium: a Web based suite of programs for comprehensive and simultaneous analysis of protein structure and sequence***.*** *Nucleic Acids Research*, v. 31, n. 13, p. 3386-3392, 2003.

PASCUAL-AHUIR, J.; SILLA, E. GEPOL: an improved description of molecular surfaces. I. Building the spherical surface set. *Journal of Computational Chemistry*, v. 11, n. 9, p. 1047-1106, Sept. 1990.

PAVLOV, M.; FEDOROV, B. Improved technique for calculating x-ray scattering intensity of biopolymers in solution: evaluation of the form volume, and surface of a particle. *Biopolymers*, v. 22, p. 1507-1522, 1983.

PERROT, G.; CHENG, B.; GILSON, K. D.; PALMER, K.; NAYEEM, A.; MAIGRETTE, B.; SCHERAGA, H. A. MSEED: a program for the rapid analytical determination of accessible surface areas and their derivates. *Journal of Computational Chemistry*, v. 13, n. 1, p. 1-11, Jan./Feb. 1992.

POWELL, M. J. D. The volume internal to three intersecting hard spheres. *Molecular Physics*, v. 7, n. 6, p. 591-592, 1963-1964.

RAY, N.; CAVIN, X.; PAUL, J. C.; MAIGRET, B. Intersurf: dynamic interface between proteins. *Journal of Molecular Graphics and Modelling,* v. 23, p. 347-354, 2005.

.

RICHARDS, F. The interpretation of protein structures: total volume, group volume distributions and packing density. *J. Mol. Biol.*, v. 82, p. 1-14, 1974.

RICHMOND, T. Solvent accessible surface area and excluded volume in proteins: Analytical equations for overlapping spheres and implications for the hydrophobic effect. *J. Mol. Biol.*, v. 178, p. 63-89, 1984.

RICHMOND, T.; RICHARDS, F. M. Packing of alpha-helices: geometrical constraints and contact areas. *J. Mol. Biol.*, v. 119, p. 537-555, 1978.

SHRAKE, A.; RUPLEY, J. Environment and exposure to solvent of protein atoms. Lisozyme and insulin. *J. Mol. Biol.*, v. 79, p. 351-371, 1973.

SUGIHARA, K. Sliver-free perturbation for the Delaunay tetrahedrization. *Computer-Aided Design,* v. 39, n. 1, p. 87-94, Feb. 2007.

WANG, H.; LEVINTHAL, C. A vectorized algorithm for calculation the accessible surface area of macromolecules. *Journal of Computational Chemistry*, v. 12, n. 7, p. 868-871, Sept. 1991.

WODAK, S.; JANIN, J. Analytical approximation to the accessible surface area of proteins. *Proc. Natl. Acad. Sci. USA*, v. 77, n. 4, p. 1736-1740, Apr. 1980

EPSTEIN, C. J.; GOLDBERGER, R. F.; ANFINSEN, C. B. The genetic control of tertiary protein structure: Studies with model systems. *Cold Spring Harbor Symp. Quant. Biol.*, v. 27, p. 439-449, 1963.

FINNEY, J. Volume occupation, environment and accessibility in proteins. The problem of protein surface. *J. Mol. Biol.*, v. 96, n. 4, p. 721-732, Aug. 1975.

GELLATLY, B.; FINNEY, J. Calculation of protein volumes: an alternative to the Voronoi procedure. *J. Mol. Biol.*, v. 161, n. 2, p. 305-322, Oct. 1982.

GERSTEIN, M.; RICHARDS, F.M. Protein geometry: volumes, areas and distances. In: ROSSMANN, M. G.; ARNOLD, E. (Ed.). *International tables for crystallography*. New York: Springer, 2006. v. F: Crystallography of biological macromolecules. Chap. 22.

GIBSON, K.; SHERAGA, H. Exact calculation of the volume and surface area of fused hard-sphere molecules with unequal atomic radii. *Molecular Physics,* v. 62, n. 5, p. 1247-1265, 1987.

GIBSON, K.; SHERAGA, H. Surface area of the intersection of three spheres with unequal radii: a simplified analytical formula. *Molecular Physics*, v. 64, n. 4, p. July, 641-644, 1988.

GRAND, S. L.; MERTZ JUNIOR, K. M. Rapid approximation to molecular surface area via the use of boolean logic and look-up tables. *Journal of Computational Chemistry,* v. 14, n. 3, p. 349-352, Mar. 1993.

HELTE, A. Fourth-order bounds on the effective conductivity for a system of fully penetrable spheres. *Proc. R. Soc. Lond*. *A: Mathematical and Physical Sciences*, v. 445, n. 1923, p. 247-256, Apr. 1994.

KRATKY, K. Intersecting disks (and spheres) and statistical mechanics. I. Mathematical basis. *J. Stat. Phys.*, v. 25, n. 4, p. 619-634, Aug. 1981.

KUNDROT, C.; PONDER, J.; RICHARDS, F. Algorithms for calculating excluded volume and its derivates as a function of molecular conformation and their use in energy minimization. *Journal of Computational Chemistry*, v. 12, n. 3, p. 402-409, Apr. 1991.

LEE, B.; RICHARDS, F.M. The interpretation of protein structures: estimation of static accessibility. *J. Mol. Biol.,* v. 55, n. 3, p. 379-400, Feb. 1971.

LEVINTHAL, C. Are there pathways for protein folding? *Extrait du Journal de Chimie Physique*, v. 65, n. 1, p. 44-45, 1968.

LIANG, J.; EDELSBRUNNER, H.; FU, P.; SUDHAKAR, P. V.; SUBRAMANIAM, S. Analytical shape computation of macromolecules: I. Molecular area and volume through alpha shapes. *Proteins*: Structure, Function, and Genetics, v. 33, p. 1-17, 1998a.

LIANG, J.; EDELSBRUNNER, H.; FU, P.; SUDHAKAR, P. V.; SUBRAMANIAM, S. Analytical shape computation of macromolecules: II. Inaccessible cavities in proteins. *Proteins*: Structure, Function, and Genetics, v. 33, p. 18-29, 1998b.

MÜCKE, E. P. */Shapes and implementations in three-dimensional geometry*. 1993. Ph.D. Thesis - Department of Computer Science, University of Illinois at Urbana-Champaign, Champaign, IL.

MULLER, J. Calculation of scattering curves for macromolecules in solution and comparison with results methods using effective atomic scattering factors. *J. Appl. Cryst.* v. 16, Part 1, p. 74-82, Feb. 1983.

Embrapa

Informática Agropecuária

Ministério da
Agricultura, Pecuária
e Abastecimento

BRASIL
UM PAÍS DE TODOS
GOVERNO FEDERAL