



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CENTRO DE CIÊNCIAS**  
**DEPARTAMENTO DE ESTATÍSTICA E MATEMÁTICA APLICADA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM E MÉTODOS**  
**QUANTITATIVOS**

**ANDERSON DA SILVA COSTA**

**ALGORITMO DE REFLEXÕES PARA O PROBLEMA DE GEOMETRIA DE**  
**DISTÂNCIAS DISCRETIZÁVEL**

**FORTALEZA**

**2020**

ANDERSON DA SILVA COSTA

ALGORITMO DE REFLEXÕES PARA O PROBLEMA DE GEOMETRIA DE DISTÂNCIAS  
DISCRETIZÁVEL

Dissertação apresentada do Programa de Pós-graduação em Modelagem e Métodos Quantitativos do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Modelagem e Métodos Quantitativos. Área de Concentração: Matemática Aplicada

Orientador: Prof. Dr. Michael Ferreira de Souza

FORTALEZA

2020

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

C87a Costa, Anderson da Silva.

Algoritmo de reflexões para o problema de Geometria de Distâncias discretizável /  
Anderson da Silva Costa. – 2020.  
58 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa  
de Pós-Graduação em Modelagem e Métodos Quantitativos, Fortaleza, 2020.

Orientação: Prof. Dr. Michael Ferreira de Souza.

1. Geometria de Distâncias. 2. Branch-and-Prune. 3. Proteínas. 4. Simetria. I. Título.

CDD 510

---

ANDERSON DA SILVA COSTA

ALGORITMO DE REFLEXÕES PARA O PROBLEMA DE GEOMETRIA DE DISTÂNCIAS  
DISCRETIZÁVEL

Dissertação apresentada do Programa de Pós-graduação em Modelagem e Métodos Quantitativos do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Modelagem e Métodos Quantitativos. Área de Concentração: Matemática Aplicada

Aprovada em:

BANCA EXAMINADORA

---

Prof. Dr. Michael Ferreira de Souza (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Jesus Ossian da Cunha  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Carlile Campos Lavor  
Universidade Estadual de Campinas (UNICAMP)

Dedico este trabalho ao meu pai, Luciano Geraldo da Costa, que sempre lutou para que seus filhos construíssem seus sonhos.

## **AGRADECIMENTOS**

Gostaria de agradecer primeiramente a Deus, por me sustentar em frente a todas as dificuldades que surgiram durante a realização deste trabalho e proporcionar a alegria de concluí-lo. Agradeço também ao meu professor e orientador Michael Ferreira de Souza que sempre me motivou a não diminuir o ritmo da pesquisa e colaborou com importantes lições acerca de como escrever bem um trabalho acadêmico.

Os agradecimentos também são direcionados aos membros da banca avaliadora, professores Jesus Ossian da Cunha e Albert Einstein F. Muritiba, membros do corpo docente do Mestrado em Modelagem e Métodos Quantitativos, e também igual gratidão aos professores externos ao programa do mestrado, Carlile Campos Lavor e Luiz Mariano Carvalho, por aceitarem fazer parte desta importante etapa de minha vida acadêmica.

"Não há assuntos pouco interessantes; apenas há  
pessoas pouco interessadas."

(G. K. Chesterton)

## RESUMO

O objetivo do Problema de Geometria de Distâncias (DGP) é determinar um conjunto de coordenadas em um espaço euclidiano que satisfaça um determinado conjunto de restrições. A aplicabilidade do DGP se espalha em diversas áreas, da geometria molecular à robótica. O algoritmo *Branch-and-Prune* (BP) é um dos algoritmos mais importantes na literatura do DGP. Basicamente, o algoritmo BP representa o espaço de busca do DGP como uma árvore binária e o reduz à resolução de uma sequência de sistemas quadráticos. Entretanto, o BP pode realizar um elevado número de operações repetidas, aumentando desnecessariamente o esforço computacional. Em vista deste problema, o presente trabalho busca apresentar uma alternativa para reduzir a quantidade de operações realizadas através de simetrias facilmente identificáveis em sua representação em grafo.

**Palavras-chave:** Geometria de Distâncias. Branch-and-Prune. Proteínas. Simetria.



## ABSTRACT

The objective of the Distance Geometry Problem (DGP) is to determine a set of coordinates in a Euclidean space that satisfies a given set of restrictions. The applicability of DGP spreads in several areas, from molecular geometry to robotics. The *Branch-and-Prune* (BP) algorithm is one of the most important algorithms in the DGP literature. Basically, the BP algorithm represents the DGP search space as a binary tree and reduces it to the resolution of a sequence of quadratic systems. However, BP can perform a large number of repeated operations, unnecessarily increasing computational effort. In view of this problem, the present work purposes an alternative to reduce the number of operations performed through symmetries easily identifiable in DGP graph representation.

**Keywords:** Distance Geometry. Branch-and-Prune. Proteins. Symmetry.

## LISTA DE FIGURAS

Figura 1 – Em uma proteína, a função e a geometria estão intimamente conectadas. . . . .	14
Figura 2 – Máquina de Ressonância Magnética Nuclear. . . . .	15
Figura 3 – Ligações peptídicas entre aminoácidos. . . . .	16
Figura 4 – Estrutura de um aminoácido. . . . .	16
Figura 5 – Sequência de aminoácidos que formam a estrutura de uma proteína. . . . .	17
Figura 6 – Flexibilidade da estrutura da proteína. . . . .	17
Figura 7 – Estrutura do sítio de ligação do <i>fosfato inositol</i> em um domínio PH. . . . .	18
Figura 8 – Transformada Gaussiana de uma função. Na esquerda a função original ( $\lambda = 0$ ) e na direita a função suavizada ( $\lambda = 0.3$ ). . . . .	23
Figura 9 – Posição de um átomo calculada através das distâncias aos átomos de base. . . . .	23
Figura 10 – Aproximação da função $\theta_i$ com a norma euclidiana. . . . .	26
Figura 11 – Ângulos de Torção. . . . .	30
Figura 12 – Árvore binária de busca associada ao $K$ DMDGP com 4 átomos e $K = 2$ . . . . .	31
Figura 13 – Interseção de três esferas para o caso $K = 3$ . . . . .	32
Figura 14 – Soluções do exemplo anterior representadas na árvore binária de busca. . . . .	34
Figura 15 – Simulação 2D de um experimento de ressonância magnética nuclear. . . . .	35
Figura 16 – Ordenação dos átomos de hidrogênio. . . . .	36
Figura 17 – Distâncias intermoleculares e ângulos de torção. . . . .	37
Figura 18 – Soluções simétricas. . . . .	41
Figura 19 – Esquematização do retrocesso de posições. . . . .	42
Figura 20 – Mediatrizes referentes as posições $x_5^{(1)}, x_5^{(2)}$ e $x_5^{(3)}$ . . . . .	43
Figura 21 – Mediatrizes referentes as posições $x_5^{(1)}, x_5^{(4)}, x_5^{(5)}, x_5^{(7)}$ e $x_5^{(8)}$ . . . . .	44
Figura 22 – Matriz de conectividade do caso bidimensional. . . . .	44
Figura 23 – Orientação para os semi-planos. . . . .	45
Figura 24 – Reflexão Bidimensional. . . . .	48
Figura 25 – Produto Vetorial. . . . .	49
Figura 26 – Posições relativas e composições dos operadores de reflexão. . . . .	50
Figura 27 – 1bdk: 15 vértices e 40 arestas. . . . .	53
Figura 28 – 1afo: 120 vértices e 518 arestas. . . . .	53
Figura 29 – 3fhh: 1863 vértices e 7796 arestas. . . . .	54

## LISTA DE TABELAS

Tabela 1 – Tempo Computacional e LDE dos Experimentos Numéricos . . . . .	54
---	----

## SUMÁRIO

1	INTRODUÇÃO . . . . .	12
2	FUNDAMENTAÇÃO TEÓRICA . . . . .	14
2.1	Sobre proteínas . . . . .	14
2.2	Geometria das proteínas . . . . .	15
2.3	Geometria de Distâncias . . . . .	18
2.3.1	<i>Formulação matemática do DGP</i> . . . . .	19
2.3.2	<i>Complexidade do DGP</i> . . . . .	20
3	MÉTODOS CONTÍNUOS PARA O DGP . . . . .	22
3.1	Algoritmo DGSOL . . . . .	22
3.2	Geometric Build-Up . . . . .	23
3.3	Suavização Hiperbólica . . . . .	25
3.4	Programação Semidefinida . . . . .	27
4	K-DGP MOLECULAR DISCRETIZÁVEL ( $^K DMDGP$ ) . . . . .	29
4.1	Formulação matemática do $^K DMDGP$ . . . . .	29
4.2	Simetria do $^K DMDGP$ . . . . .	32
4.3	Ordenação nas Proteínas . . . . .	35
4.4	Algoritmo <i>Branch-and-Prune</i> . . . . .	36
4.4.1	<i>Matrizes de Torção</i> . . . . .	36
4.4.2	<i>Estrutura do Branch-and-Prune</i> . . . . .	39
5	METODOLOGIA . . . . .	41
5.1	Symmetric Branch-and-Prune . . . . .	42
5.2	Reflexão geométrica . . . . .	47
6	RESULTADOS . . . . .	52
7	CONCLUSÕES E TRABALHOS FUTUROS . . . . .	55
	REFERÊNCIAS . . . . .	56

## 1 INTRODUÇÃO

A Geometria de Distâncias (**DG** – *Distance Geometry*) é uma área de pesquisa que mescla conceitos matemáticos e da Ciência da Computação como, por exemplo, norma, grafos, combinatória, algoritmos e teoria da complexidade. Dentre os problemas de interesse da DG, destaca-se o Problema de Geometria de Distâncias (**DGP** - *Distance Geometry Problem*) que consiste na determinação das coordenadas de um conjunto de pontos em um dado espaço geométrico, onde apenas algumas das distâncias entre estes pontos são conhecidas *a priori* (LAVOR *et al.*, 2017).

O DGP possui aplicação em diversos ramos, dentre os quais podemos citar: na astronomia, para o cálculo das coordenadas das estrelas e de outros corpos celestes (LINDEGREN *et al.*, 2012); na bioquímica, para a determinação da estrutura tridimensional de moléculas proteicas (LAVOR *et al.*, 2017); na robótica, no movimentos dos braços de uma robô (NIELSEN; ROTH, 1999); e na telecomunicações, no posicionamento de sensores wireless (EREN *et al.*, IEEEExplore, 2004).

Existem diversas variações do DGP (LAVOR; LIBERTI, 2014). Neste projeto, consideraremos a variante denominada de *K-Problema Discretizável de Geometria de Distâncias Moleculares* ( $K$ DMDGP) cujo espaço de solução é discreto. Com mais de 150 citações<sup>1</sup>, o algoritmo Branch-and-Prune (BP) é um dos mais relevantes para a solução do  $K$ DMDGP. Basicamente, o BP representa o espaço de busca do  $K$ DMDGP como uma árvore binária e reduz a busca de solução a uma sequência de problemas quadráticos ordenados por uma busca em profundidade (LIBERTI *et al.*, 2008).

O objetivo principal deste trabalho consiste em apresentar uma nova abordagem para solucionar o problema de Geometria de Distâncias. Esta nova abordagem utiliza simetrias a fim de obter uma redução no esforço computacional sem reduzir a qualidade das soluções obtidas. Os detalhes sobre as propriedades utilizadas serão apresentadas e discutidas no decorrer deste trabalho, com detalhes específicos nas referências citadas.

Além do objetivo anterior, a motivação para o estudo deste problema diz respeito a sua dificuldade, pois o DGP é um problema NP-Difícil (SAXE, 1980) e a escassez de distâncias (restrições) faz com que o DGP torne-se um problema ainda mais desafiador.

O trabalho está organizado em 7 capítulos, onde o primeiro corresponde exatamente a esta introdução acerca do Problema de Geometria de Distâncias. No capítulo 2 são apresentadas

<sup>1</sup> Número de citações segundo o Google Scholar em 01/03/2020.

a base teórica e alguns métodos para o Problema de Geometria de Distâncias com foco em sua aplicação no cálculo da estrutura tridimensional de proteínas. Nos capítulos 3 e 4 dá-se destaque para duas variações do DGP, sendo uma delas específica para o cálculo de distâncias moleculares. É também apresentado o algoritmo *Branch-and-Prune*, o mais tradicional da literatura do DGP. Nos capítulos 5 e 6 são apresentados, respectivamente, a proposta deste trabalho e os resultados numéricos obtidos nos testes computacionais, onde comparamos nossa proposta com o algoritmo *Branch-and-Prune*. Finalmente, no capítulo 7 são apresentadas as conclusões e as propostas de trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

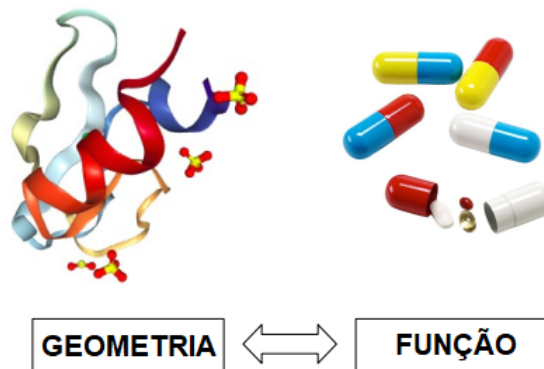
Este capítulo motiva o estudo do DGP. Nele serão conceituadas sua definição e variantes, algumas de suas aplicações e também variados métodos de solução.

### 2.1 Sobre proteínas

As proteínas são componentes fundamentais para o bom funcionamento de um organismo vivo. Dentre as funções das proteínas, podemos citar o transporte de oxigênio, a proteção contra patógenos e a catalisação de reações (NEUMAIER, 1997).

As propriedades e a função de cada proteína estão associadas a sua estrutura tridimensional (Figura 1) (LAVOR *et al.*, 2017), que consiste na disposição de seus aminoácidos, esta estrutura também afeta a forma como as proteínas interagem com outras substâncias como, por exemplo, medicamentos. Isto faz com que a determinação da estrutura tridimensional seja um problema de suma importância para a indústria farmacêutica, visto o elevado custo de seus procedimentos (SCHLICK, 2010).

Figura 1 – Em uma proteína, a função e a geometria estão intimamente conectadas.



Para determinar tal estrutura, os métodos empíricos mais tradicionais são a Cristalografia de Raios-X e a Ressonância Magnética Nuclear (RMN) (DONG; WU, 2002). A primeira trabalha com a análise da difração produzida por um feixe de raios-X incidido diretamente sobre um cristal de qualidade adequada (SCHLICK, 2010). A detecção e o escaneamento fornecem um mapa tridimensional de densidade eletrônica, que consiste basicamente numa disposição das prováveis posições dos átomos da estrutura molecular. Sua principal dificuldade está na obtenção de cristais com a qualidade desejada.

Já a técnica de RMN não fornece um detalhamento como a cristalografia, porém

ela é capaz de preservar características fisiológicas das macromoléculas e fornece informações não estáticas e que incorporam efeitos térmicos (SCHLICK, 2010). Os dados fornecidos pela RMN consistem em distâncias (curtas) euclidianas entre átomos de hidrogênio em uma proteína (LAVOR *et al.*, 2019). Através desse conjunto parcial de distâncias, formula-se o problema de determinação da estrutura tridimensional de proteínas.

Vale ressaltar que não existe uma competição entre estes métodos pois ambos fornecem informações limitadas e imprecisas, sendo portanto complementares. Combinados a algumas características da geometria das proteínas, os dados produzidos por ambos os métodos auxiliam na formulação matemática do problema da estrutura molecular.

Figura 2 – Máquina de Ressonância Magnética Nuclear.



Fonte: (LAVOR *et al.*, 2017)

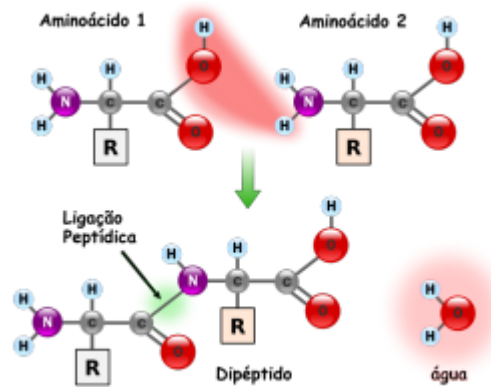
Algumas das estruturas tridimensionais obtidas experimentalmente ou por desenvolvimento teórico estão armazenados no repositório PDB (*Protein Data Bank*) (NATIONAL... , 2020), sendo este uma importante fonte de informação e conhecimento acerca das proteínas (BERMAN *et al.*, 2000).

## 2.2 Geometria das proteínas

As proteínas são macromoléculas geradas pela combinação de até 20 diferentes tipos de aminoácidos, dispostos sequencialmente e fixados por ligações peptídicas (Figura 3) (DEVLIN, 1998). A disposição dos átomos de cada proteína é bem particular e caracteriza sua função.



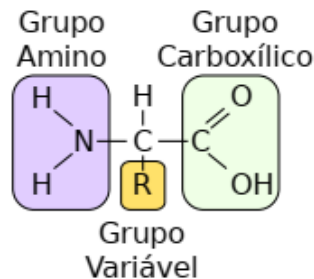
Figura 3 – Ligações peptídicas entre aminoácidos.



Fonte: (CAMARGO, 2015)

Dentro da estrutura atômica de uma proteína existe uma subestrutura específica chamada de cadeia principal. A cadeia principal é uma sequência formada por um átomo de nitrogênio (N) e dois carbonos (C), sendo um dos carbonos chamado de  $C_{\alpha}$ , ao qual ligam-se quatro diferentes grupos: um átomo de hidrogênio; um grupo amino; um grupo ácido carboxílico e uma cadeia variável chamada de *lateral* (Figura 4).

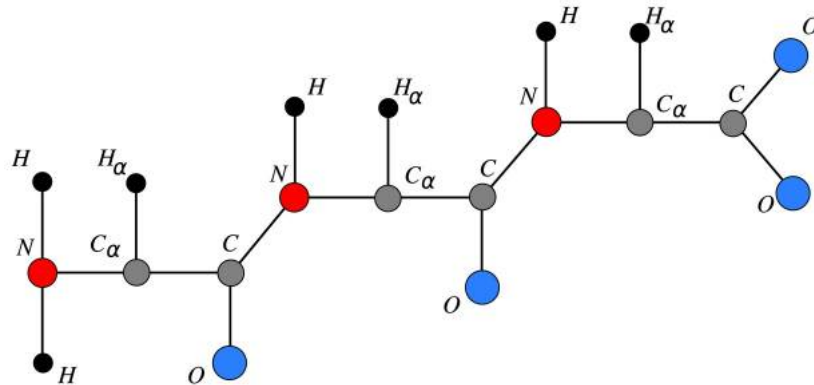
Figura 4 – Estrutura de um aminoácido.



Fonte: (LAVOR *et al.*,)

Sendo assim, uma proteína é uma sequência de resíduos de aminoácidos unidos por ligações peptídicas sucessivas. Cada proteína é caracterizada de modo único pela sequência de aminoácidos de sua estrutura (Figura 5). Assim como no alfabeto latino formado por 23 letras, podemos construir inúmeras palavras, no alfabeto proteico, os 20 aminoácidos estão para as proteínas assim como as 23 letras do alfabeto latino estão para palavras da língua portuguesa.

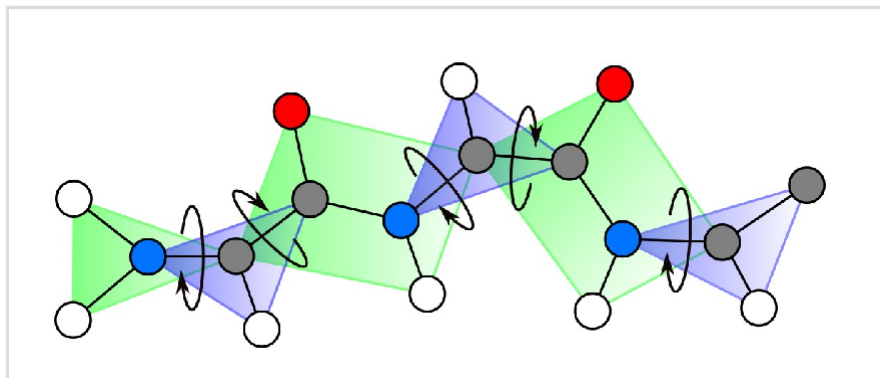
Figura 5 – Sequência de aminoácidos que formam a estrutura de uma proteína.



Fonte: (LAVOR *et al.*, 2019)

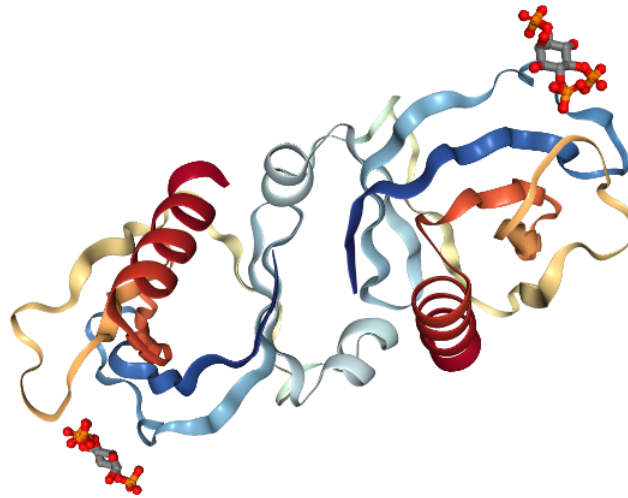
As ligações peptídicas consistem no encontro entre o grupo amino de um aminoácido e o grupo carboxílico de outro. No entanto, são as ligações associadas ao  $C_{\alpha}$  que produzem a flexibilidade da estrutura da proteína pois, pela presença de forças interatômicas, a estrutura é torcida e encurvada gerando assim diversas conformações distintas (SOUZA, 2010). A Figura 6 ilustra os graus de liberdade associados a rotações com eixos nas ligações  $N - C_{\alpha}$  e  $C_{\alpha} - C$ .

Figura 6 – Flexibilidade da estrutura da proteína.



Esta “dobradura” e a ordem em que os aminoácidos estão dispostos são o que caracteriza a estrutura tridimensional da proteína e conseqüentemente definem a sua funcionalidade. Vale ressaltar que a quantidade de átomos presentes nas proteínas varia de 58 até 27000 átomos, porém a grande maioria possui número de átomos na casa das centenas. Na Figura 7, vemos a representação de uma proteína com 888 átomos.

Figura 7 – Estrutura do sítio de ligação do *fosfato inositol* em um domínio PH.



Fonte: <https://www.rcsb.org/structure/1BTN> (NATIONAL... , 2020)

### 2.3 Geometria de Distâncias

A Geometria de Distâncias (do inglês DG – *Distance Geometry*) é uma área de pesquisa que utiliza conceitos da Matemática (norma, grafo, combinatória, ...) e da Ciência da Computação (algoritmos e teoria da complexidade). Seu problema fundamental é determinar as coordenadas (cartesianas) de um conjunto de pontos em um determinado espaço geométrico (*euclidiano*) através das distâncias entre estes pontos (LAVOR *et al.*, 2017). O desafio ou principal dificuldade desta área ocorre quando há escassez de distâncias, visto que a situação onde todas são conhecidas é facilmente resolvida por pequenos problemas de trilateração (DONG; WU, 2003).

A origem da DG data de 1928 quando Menger caracterizou diversos conceitos geométricos em termos de distância (MENGER, 1930). Contudo, a consolidação da Geometria de Distâncias como uma nova área de conhecimento só veio a ocorrer em 1953 com Blumenthal (BLUMENTHAL, 1953. 347p). O grande marco histórico da área deu-se com o trabalho de Crippen and Havel acerca da aplicação do DG no cálculo de estruturas moleculares (CRIPPEN *et al.*, 1988). Desde então, este tópico de pesquisa passou a despertar o interesse de práticos pela sua aplicabilidade em diferentes áreas (astronomia, bioquímica, telecomunicações, robótica, etc...) e também de pesquisadores pela sua vasta e rica teoria, tornando-a assim uma área consolidada da matemática aplicada (LIBERTI *et al.*, 2014).

### 2.3.1 Formulação matemática do DGP

A formulação do Problema de Geometria de Distâncias (do inglês DGP - *Distance Geometry Problem*) utiliza a noção de grafo. Este objeto matemático será essencial por permitir explorar as propriedades do problema e além disto, simplifica consideravelmente a formulação. Esta representação abstrata permite que os vértices do grafo sejam qualquer objeto (estrelas, átomos, pessoas,...) e suas arestas qualquer relação entre estes objetos, justificando assim a sua vasta aplicabilidade (LAVOR *et al.*, 2017). A definição formal do DGP é dada por

**Definição:** Sejam  $K > 0$  um inteiro e  $G = (V, E, d)$  um grafo não-direcionado, onde  $V$  é o conjunto de vértices,  $E \subset V \times V$  é o conjunto de arestas e  $d : E \rightarrow (0, \infty)$  é uma função peso que representa o valor da distância entre um par de vértices cuja aresta pertence a  $E$ . O DGP consiste em determinar uma função  $x : V \rightarrow \mathbb{R}^K$  tal que:

$$\|x(u) - x(v)\| = d(\{u, v\}), \quad \forall \{u, v\} \in E \quad (2.1)$$

Fornecer uma solução para este problema significa associar cada vértice de  $G$  a um único ponto do espaço de interesse  $\mathbb{R}^K$ , de tal modo que as distâncias coincidam com os dados fornecidos pela função peso. A função  $x$  que mapeia os vértices de um grafo no espaço  $\mathbb{R}^K$  é chamada *realização* do grafo. Caso a realização satisfaça todas as equações 2.1, esta é chamada de *realização válida* (LIBERTI *et al.*, 2014). Se  $H$  é um subgrafo de  $G$  e  $x$  é uma realização válida de  $H$  em  $\mathbb{R}^K$  (considerando apenas as restrições em  $H$ ), diz-se que  $x$  é uma *realização parcial válida* de  $G$ , este conceito será importante para a definição das variantes do DGP.

A fim de simplificar a notação, chamaremos  $x(u)$  e  $x(v)$ , de  $x_u$ ,  $x_v$ , e  $d(\{u, v\})$  de  $d_{uv}$ . Além disso, como existe uma relação biunívoca entre as arestas em  $E$  e as restrições do DGP, usaremos a expressão "resolver a aresta  $\{u, v\}$ " no lugar de "resolver a restrição associada à aresta  $\{u, v\}$ ". Uma instância DGP será representada pelo par  $(G, K)$ .

Na aplicação aqui adotada para o DGP, cada vértice corresponde a um ponto no espaço geométrico tridimensional, ou seja, cada vértice será um vetor de três componentes

$$x_i = (x_{i_1}, x_{i_2}, x_{i_3})^T \quad (2.2)$$

Caso a distância  $\|x_i - x_j\|_2$  entre um dado par de vértices  $x_i$  e  $x_j$  seja conhecida, haverá uma aresta conectando-os no grafo.

### 2.3.2 Complexidade do DGP

Apesar do DGP ser formulado utilizando Teoria dos Grafos, podemos reformulá-lo através de programação matemática, sendo, neste caso, tratado como o problema de otimização irrestrita

$$\min_x f(x) = \sum_{\{u,v\} \in E} (||x_u - x_v|| - d_{uv})^2, \quad (2.3)$$

onde  $x : V \rightarrow \mathbb{R}^K$ .

É imediato que, para qualquer solução  $\bar{x}$  do DGP, teremos  $f(\bar{x}) = 0$ , sendo  $\bar{x}$  portanto um minimizador global de  $f$ . Logo a formulação de otimização ou a dada em 2.1 são equivalentes quando o conjunto solução do DGP não é vazio. Apesar da facilidade em caracterizar os minimizadores globais, essa formulação de otimização apresenta como obstáculo a grande quantidade de minimizadores locais da função  $f$  que não são globais. Na verdade, como destacado em (LAVOR *et al.*, 2017), o número de minimizadores locais que não são globais cresce exponencialmente com o número de vértices e arestas usados na definição do DGP. No cálculo de estruturas moleculares, que podem ser formadas por até 27 mil átomos, o elevado número de minimizadores locais que não são solução inviabiliza a aplicação direta de métodos de otimização não-linear. Uma das formas de reduzir o esforço computacional na otimização desta função é suavizá-la, ou seja, buscar alternativas para reduzir a quantidade de minimizadores locais (MORÉ; WU, 1999; SOUZA *et al.*, 2011). Nestas estratégias, os algoritmos de otimização clássicos são aplicados em versões modificadas (transformadas) da função  $f$ .

Por outro lado, mesmo a formulação do DGP como um sistema de equações não-lineares apresentada em 2.1 não pode ser diretamente resolvida (BAJAJ, 1988). Na realidade, resolver o DGP não é uma tarefa trivial, pois tanto no caso geral (SAXE, 1980), como também em sua variante molecular (LAVOR *et al.*, 2012), o DGP é um problema NP-difícil, ou seja, existem instâncias difíceis de resolver mesmo utilizando supercomputadores.

Caso não seja vazio, o conjunto solução do DGP sempre será infinito não-enumerável. De fato, se  $x$  é solução, então  $y = R(x)$  com  $R$  sendo uma isometria também será solução pois  $||y_u - y_v|| = ||x_u - x_v||$  (propriedade das simetrias). Para evitar a redundância na representação do conjunto solução, são considerados conjuntos solução mínimos formados apenas por soluções que não podem ser obtidas uma das outras por mera aplicação de isometrias. Neste caso, ainda podemos ter um conjunto não enumerável de soluções, pois as estruturas podem ser flexíveis, ou

seja, o conjunto de restrições consideradas podem permitir deformações contínuas em um dada solução.

### 3 MÉTODOS CONTÍNUOS PARA O DGP

Neste capítulo serão apresentados alguns dos mais conhecidos métodos presentes na literatura do DGP, onde todos são baseados em buscas no espaço Euclidiano contínuo.

#### 3.1 Algoritmo DGSOL

Proposto por (MORÉ; WU, 1999), o algoritmo DGSOL realiza uma suavização da função  $f$ , ou seja, busca reduzir a quantidade de minimizadores locais, para assim aplicar um algoritmo de otimização não-linear na função transformada. Uma sequência de minimizadores da função transformada é gerada, onde o último termo é o candidato a minimizador global da função  $f$  (MORÉ; WU, 1999). Tal estratégia concentra-se nas regiões mais promissoras com minimizadores locais profundos.

A suavização da função  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  é dada pela transformada gaussiana, gerando uma nova função

$$\langle f \rangle_\lambda = \frac{1}{\pi^{n/2} \lambda^n} \int_{\mathbb{R}^n} f(y) e^{-\frac{\|y-x\|^2}{\lambda^2}} dy \quad (3.1)$$

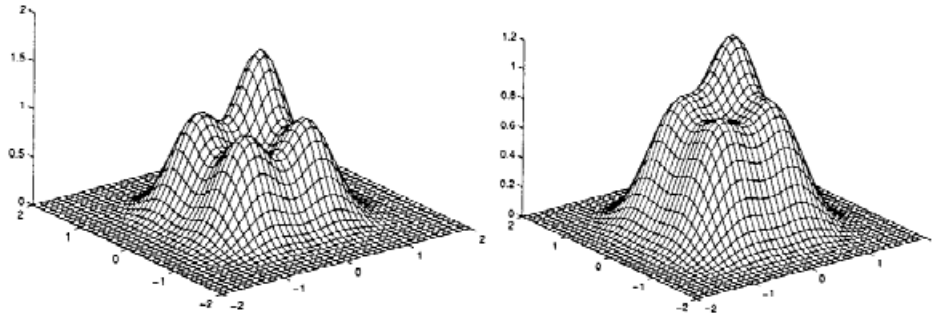
onde  $\lambda$  é o parâmetro de suavização. A função  $\langle f \rangle_\lambda$  consiste em uma média ponderada da função  $f$  em uma vizinhança de  $x$  cujo tamanho é controlado pelo parâmetro de suavização  $\lambda$  (Figura 8).

Além disto, temos a seguinte propriedade:

$$\lim_{\lambda \rightarrow 0} \langle f \rangle_\lambda (x) = f(x), \quad (3.2)$$

ou seja, para  $\lambda = 0$  a função suavizada retorna a função original. Este parâmetro  $\lambda$  é muito importante pois convexifica a função transformada, entretanto uma má escolha deste pode degenerar a solução obtida. De fato, valores grandes de  $\lambda$  fazem com que as informações da função original sejam perdidas e valores pequenos pouco reduzem a quantidade de minimizadores. Portanto, o parâmetro  $\lambda$  deve ser escolhido cautelosamente.

Figura 8 – Transformada Gaussiana de uma função. Na esquerda a função original ( $\lambda = 0$ ) e na direita a função suavizada ( $\lambda = 0.3$ ).



Fonte: (MORÉ; WU, 1999)

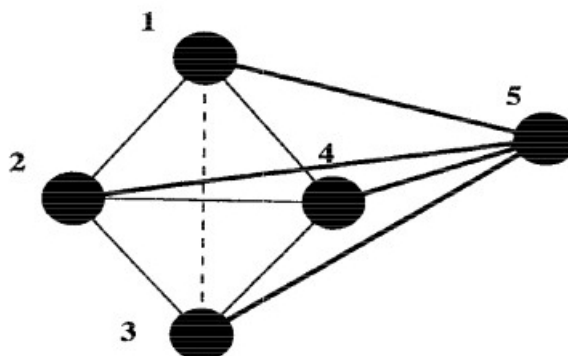
Em (MORÉ; WU, 1999), o DGSOL foi testado para instâncias entre 100 e 200 átomos, sendo capaz de resolvê-las independentemente do ponto inicial. No entanto, estas instâncias não condizem com a estrutura nem com a magnitude dos problemas reais.

### 3.2 Geometric Build-Up

Proposto por (DONG; WU, 2003), o algoritmo Geometric Build-Up baseia-se em trilaterações sucessivas: as coordenadas de cada vértice são calculadas utilizando as coordenadas e as distâncias a vértices previamente fixados.

Quando todas as distâncias são conhecidas, o Geometric Build-Up é capaz de fornecer uma solução para o DGP em  $O(n)$  operações através de simples e pequenos sistemas de equações algébricas (DONG; WU, 2003). O algoritmo inicia fixando quatro vértices para os quais existe uma clique no grafo de definição do DGP. Em seguida, a cada iteração, um novo vértice é fixado por uma trilateração utilizando quatro vértices anteriormente fixados que lhe sejam adjacentes e não coplanares (Figura 9).

Figura 9 – Posição de um átomo calculada através das distâncias aos átomos de base.



Fonte: (DONG; WU, 2003)



Tomando como exemplo o caso tridimensional e supondo conhecida a clique, as coordenadas dos quatro primeiros vértices são obtidas rapidamente centrando o primeiro deles na origem do espaço geométrico de interesse, ou seja,

$$(u_1, v_1, w_1) = (0, 0, 0), \quad (3.3)$$

os demais vértices devem ser posicionados de modo a garantir que os quatro não sejam coplanares. Por exemplo, podemos tomar

$$(u_2, v_2, w_2) = (d_{12}, 0, 0)$$

$$(u_3, v_3, w_3) = ((d_{13}^2 - d_{23}^2 + u_2^2)/(2u_2), \pm(d_{13}^2 + u_2^2)^{1/2}, 0),$$

$$(u_4, v_4, w_4) = ((d_{14}^2 - d_{24}^2 + u_2^2)/(2u_2), (d_{24}^2 - d_{34}^2 - (u_4 - u_2)^2 + (u_4 - u_3)^2 + v_3^2)/(2v_3), \\ \pm(d_{24}^2 - u_4^2 - v_4^2)^{1/2})$$

Os demais vértices são posicionados através da resolução de sistemas lineares. Por exemplo, se  $x_1, x_2, x_3$  e  $x_4$  em  $\mathbb{R}^3$  são as coordenadas não-coplanares dos vértices  $v_1, v_2, v_3$  e  $v_4$ , cujas distâncias ( $d_{v_j, v_i} = d_{j,i}, \forall i \in \{1, 2, 3, 4\}$ ) para um vértice  $v_j$  ainda não determinado são todas conhecidas, teremos as restrições

$$\|x_j - x_1\| = d_{j,1},$$

$$\|x_j - x_2\| = d_{j,2},$$

$$\|x_j - x_3\| = d_{j,3},$$

$$\|x_j - x_4\| = d_{j,4}.$$

Elevando ao quadrado ambos os lados de cada equação anterior, temos

$$\|x_j\|^2 - 2 \cdot x_j^T \cdot x_1 + \|x_1\|^2 = d_{j,1}^2,$$

$$\|x_j\|^2 - 2 \cdot x_j^T \cdot x_2 + \|x_2\|^2 = d_{j,2}^2,$$

$$\|x_j\|^2 - 2 \cdot x_j^T \cdot x_3 + \|x_3\|^2 = d_{j,3}^2,$$

$$\|x_j\|^2 - 2 \cdot x_j^T \cdot x_4 + \|x_4\|^2 = d_{j,4}^2.$$

Ao subtrair uma destas equações das demais, obtém-se um sistema linear que pode ser utilizado, caso possua solução, para determinar as coordenadas do vértice  $x_j$  (LIBERTI *et al.*, 2008). Por exemplo, ao subtrairmos a primeira delas das outras, obtemos

$$A \cdot x_j = b, \quad (3.4)$$

onde

$$A = -2 \cdot \begin{bmatrix} (x_1 - x_2)^T \\ (x_1 - x_3)^T \\ (x_1 - x_4)^T \end{bmatrix}$$

e

$$b = \begin{bmatrix} (d_{j,1}^2 - d_{j,2}^2) - (\|x_1\|^2 - \|x_2\|^2) \\ (d_{j,1}^2 - d_{j,3}^2) - (\|x_1\|^2 - \|x_3\|^2) \\ (d_{j,1}^2 - d_{j,4}^2) - (\|x_1\|^2 - \|x_4\|^2) \end{bmatrix}.$$

Quando o conjunto de distâncias é incompleto, ou seja, o grafo  $G$  não é completo, cada vértice pode requerer uma base distinta. Mais ainda, a depender do conjunto (clique) inicial, pode ser impossível fixar todos os vértices. Por isso, a primeira etapa do Geometric Build-Up é a obtenção de uma ordem válida para a fixação dos vértices. Neste caso, para grafos incompletos, a complexidade do Geometric Build-Up é  $O(n^4)$  (DONG; WU, 2003).

### 3.3 Suavização Hiperbólica

Os modelos via programação matemática (equação 2.3) frequentemente utilizados para o DGP apresentam dois obstáculos que dificultam a aplicação dos métodos de otimização não-linear: a não diferenciabilidade e a grande quantidade de minimizadores locais que não são globais (SOUZA *et al.*, 2011). O primeiro obstáculo procede da presença da norma Euclidiana na função objetivo, já a segunda se deve ao fato da natureza combinatória do problema, cuja quantidade de minimizadores aumenta a medida que as quantidades de vértices e arestas crescem (LAVOR; LIBERTI, 2014).

A técnica da suavização hiperbólica procura tanto introduzir a diferenciabilidade nos modelos formulados via programação matemática, como também reduzir a quantidade de minimizadores, através do controle de determinados parâmetros.

A diferenciabilidade será obtida pela substituição da norma Euclidiana  $\|\cdot\|$  pela seguinte função:

$$\theta_t(x_i) = \sqrt{t^2 + \langle x_i, x_i \rangle}, \quad (3.5)$$

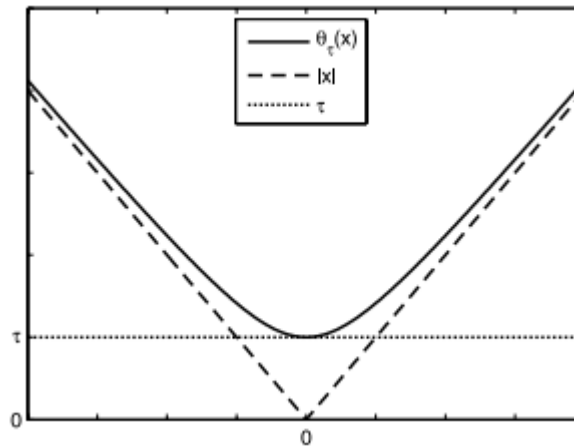
onde  $\langle \cdot \rangle$  representa o produto interno e  $t > 0$ .

A função  $\theta_t(x_i)$  possui as seguintes propriedades (SOUZA *et al.*, 2011):

1.  $\lim_{t \rightarrow 0} \theta_t(x_i) = \|x_i\|$ , ou seja, a função é uma boa aproximação para a norma Euclidiana;
2.  $\theta_t$  é infinitamente diferenciável;
3.  $\theta_t(x_i) > \|x_i\|, \forall x_i$ ;
4.  $t_1 > t_2 \rightarrow \theta_{t_1}(x_i) > \theta_{t_2}(x_i), \forall x_i \in \mathbb{R}^n$ .

O parâmetro  $t$  é o responsável por aproximar a função  $\theta_t$  da norma Euclidiana, conforme a Figura 10.

Figura 10 – Aproximação da função  $\theta_t$  com a norma euclidiana.



Fonte: (SOUZA *et al.*, 2011)

Aplicando-a na função objetivo 2.3 temos

$$f(x) = \sum (\theta_t(x_u - x_v) - d_{uv})^2. \quad \forall \{u, v\} \in E \quad (3.6)$$

Esta nova função objetivo pertence à classe  $C^\infty$ , permitindo que os métodos clássicos possam ser utilizados, entretanto ela difere da função objetivo do problema original (equação 2.3). A fim de obter a solução de 2.3 é proposta então a resolução de uma sequência de problemas

parametrizados (SOUZA *et al.*, 2011), onde a sequência das soluções obtidas tenderá, no limite, à solução do problema original (SOUZA, 2010).

### 3.4 Programação Semidefinida

Proposto por (BISWAS, 2007), o algoritmo via programação semidefinida envolve uma reformulação e uma relaxação do problema de geometria de distâncias, que consiste em

$$\begin{aligned} & \text{Determinar } X \\ & \text{s.a. } \|x_i - x_j\|^2 = d_{ij}^2 \end{aligned}$$

sendo  $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{K \times n}$  a matriz formada pelas coordenadas de todos os  $n$  vértices  $x_i \in \mathbb{R}^K$ . Tal formulação pode ser reescrita como (BISWAS *et al.*, 2008)

$$\begin{aligned} & \text{Determinar } Y \\ & \text{s.a. } e_{ij}^t Y e_{ij} = d_{ij}^2 \\ & Y = X^t X \end{aligned}$$

onde  $Y$  é a chamada *Matriz de Gram* ou *Matriz do Produto Interno* e  $e_{ij} = e_i - e_j$ , sendo  $e_i \in \mathbb{R}^n$ , o  $i$ -ésimo vetor da base canônica. As restrições desta reformulação equivalem às originais, pois

$$e_{ij}^t Y e_{ij} = e_{ij}^t X^t X e_{ij} = (x_i - x_j)^t (x_i - x_j) = \|x_i - x_j\|^2. \quad (3.7)$$

Entretanto, pode não ser possível decompor a matriz  $Y$  no produto  $X^t X \in \mathbb{R}^{K \times n}$ . Para este caso, em (BISWAS *et al.*, 2008), é apresentada a relaxação

$$\begin{aligned} & \text{Determinar } \text{tr}(Y) \\ & \text{s.a. } e_{ij}^t Y e_{ij} = d_{ij}^2 \\ & Y \succcurlyeq 0 \end{aligned}$$

A função objetivo desta relaxação é equivalente ao somatório do quadrado das normas, isto ocorre pela possibilidade de, sem perda de generalidade, fixar o centro de gravidade dos pontos  $x_i$  na origem do espaço de interesse. Desta forma, não existem impedimentos em substituir  $\text{tr}(Y)$  pelo  $\sum \|x_i\|^2$  (SOUZA, 2010).

A segunda restrição advém de um importante resultado que afirma que existe uma realização válida na dimensão  $K$  se, e somente se, a matriz de Gram  $Y$  é semidefinida positiva e tenha posto igual a  $K$  (BISWAS *et al.*, 2008). Nos casos em que há dados faltantes faz-se

necessário completar a matriz de Gram do problema, a fim de garantir que uma realização na dimensão  $K$  seja possível.

A aplicação desta estratégia também possui como ideia principal a *clusterização* dos vértices do grafo associado ao problema com que cada *cluster* correspondendo a um subgrafo (LIBERTI *et al.*, 2011). Esta estratégia obtém realizações parciais válidas para cada subgrafo e, em seguida, as combina, obtendo coordenadas globais para todos os vértices do grafo  $G$ .

## 4 K-DGP MOLECULAR DISCRETIZÁVEL ( $K$ DMDGP)

Após a apresentação dos métodos mais conhecidos para a abordagem contínua do DGP, discutiremos a variação discreta do problema. Por meio dela será possível definir a variante do DGP conhecida por *K-Problema Discretizável de Geometria de Distâncias Moleculares* ( $K$ DMDGP), cujo termo "molecular" refere-se a sua principal aplicação: a conformação de moléculas proteicas.

### 4.1 Formulação matemática do $K$ DMDGP

O  $K$ DMDGP surgiu do interesse no cálculo da estrutura tridimensional de proteínas, sendo posteriormente generalizado para o espaço  $K$ -dimensional (CAMARGO, 2015). A definição formal do  $K$ DMDGP, de acordo com (GONCALVES *et al.*, ), é dada por

**Definição:** Uma instância DGP  $(G, K)$  é um  $K$ DMDGP se existe uma ordenação dos vértices de  $G$ , denotada por  $\{v_1, v_2, \dots, v_n\}$ , tal que:

- $G[\{v_1, v_2, \dots, v_K\}]$  é uma clique, com  $K < n$ , onde  $K$  é a dimensão do espaço de interesse;
- Para todo  $i > K$ ,  $v_i$  é adjacente  $v_{i-1}, \dots, v_{i-K}$ ;
- $CM(v_{i-1}, \dots, v_{i-K}) \neq 0$ ;

onde  $G[\cdot]$  denota o subgrafo induzido e  $CM(v_{i-1}, \dots, v_{i-K})$  representa o determinante de Cayley-Menger de  $v_{i-1}, \dots, v_{i-K}$  (GONCALVES *et al.*, ).

O  $K$ DMDGP consiste em determinar uma função  $x: V \rightarrow \mathbb{R}^K$  tal que:

$$\|x_{v_i} - x_{v_j}\| = d_{i,j}, \quad \forall \{v_i, v_j\} \in E. \quad (4.1)$$

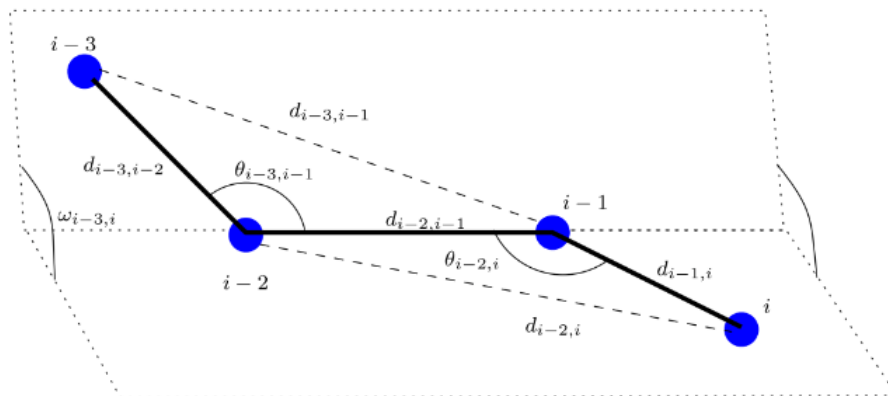
Da mesma forma que em 2.3.1, a função  $x$  é chamada de *realização* de  $G$  e aquelas que atendem a todas as equações 4.1 são chamadas de *válidas*.

A condição de existência das cliques, que no caso tridimensional é representada por  $\{v_i, v_{i-1}, v_{i-2}, v_{i-3}\}$ ,  $\forall i = \{4, \dots, n\}$ , fornece mais informações sobre a ordenação dos vértices do grafo molecular. De acordo com (LAVOR *et al.*, 2017), tais informações são:

1. As distâncias  $d_{ij}$  entre três vértices consecutivos;
2. Os ângulos planos  $\theta_{ij}$  entre três vértices consecutivos;
3. Os ângulos de torção  $\omega_{ik}$  entre quatro vértices consecutivos.

Os ângulos de torção são os responsáveis por estabelecer a estrutura tridimensional do grafo molecular (Figura 11), no entanto, o  $KDMDGP$  possui apenas a informação do cosseno dos ângulos de torção, desta forma, cada um desses ângulos pode assumir dois valores no intervalo  $[0, 2\pi]$ .

Figura 11 – Ângulos de Torção.

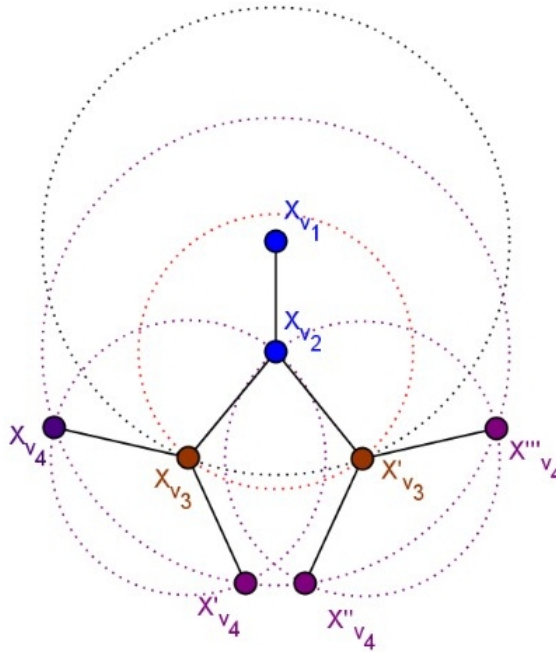


Fonte: (LAVOR *et al.*, 2012)

Uma solução do  $KDMDGP$  pode ser obtida a partir da resolução de uma sequência de sistemas quadráticos. Tais sistemas são construídos utilizando  $K$  vértices anteriores que estejam conectados aos que serão realizados. Todavia, qualquer procedimento de resolução também deve levar em consideração as arestas adicionais que possam existir. Estas arestas adicionais são utilizadas nos chamados *testes de viabilidade* e permitem excluir realizações que não atendam a estrutura completa.

Graças ao conjunto de hipóteses da formulação do  $KDMDGP$ , cada vértice do grafo  $G$  poderá ocupar um número finito de posições no espaço  $K$ -dimensional, atendendo as restrições associadas aos demais vértices (CAMARGO, 2015). Isto faz com que o espaço de busca torne-se discreto e, mais ainda, possa ser representado por uma árvore binária (Figura 12) (LAVOR *et al.*, 2017). Os  $K$  primeiros vértices são realizados em uma etapa de pré-processamento e as possíveis posições dos vértices são representadas em cada nível da árvore (Figura 12). As soluções do problema, apresentam-se agora como caminhos da raiz da árvore de busca até o seu nível mais inferior (folhas) (LAVOR; LIBERTI, 2014).

Figura 12 – Árvore binária de busca associada ao  $KDMDGP$  com 4 átomos e  $K = 2$ .



Fonte: (CAMARGO, 2015)

A largura da árvore de busca associada ao  $KDMDGP$  cresce exponencialmente com o número de vértices, portanto explorá-la como um todo é proibitivo. Para prevenir esta situação, é essencial remover as regiões de inviabilidade, reduzindo assim o espaço de busca. Para tal, particiona-se o conjunto de arestas em dois subconjuntos de suma importância para a resolução do  $KDMDGP$ : as *arestas de discretização* e as *arestas de poda* (LAVOR; LIBERTI, 2014).

As *arestas de discretização*  $E_D$  são as responsáveis pela construção da árvore de busca, enquanto que as *arestas de poda*  $E_P$  tem como função remover as regiões de inviabilidade. Cada sistema quadrático é construído a partir dos  $K$  vértices anteriores ao vértice  $v_i$  a ser realizado. Geometricamente, cada sistema representa a interseção de  $K$  hiperesferas, ou seja,

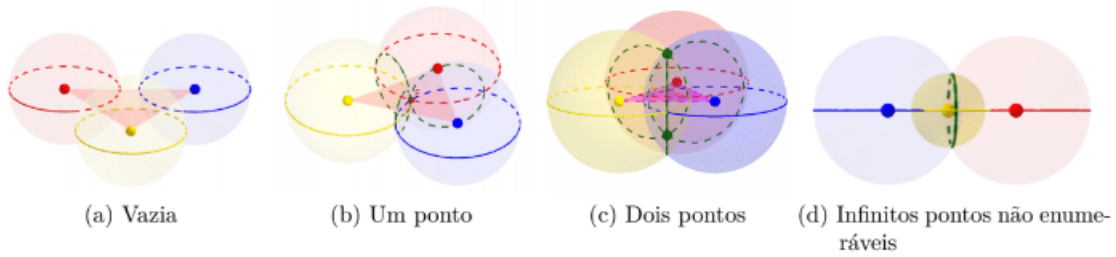
$$D = \bigcap_{p=1}^K S(x_{v_{i-p}}, d_{v_{i-p}v_i}), \quad (4.2)$$

onde  $v_{i-p}$  são todos os vértices anteriores a  $v_i$  (Figura 13). Caso a cardinalidade deste conjunto interseção seja nula, não haverá realização viável para um dado vértice  $v_i$ .

Os testes de viabilidade serão os responsáveis, através das arestas de poda, por selecionar quais dentre as realizações possíveis, obtidas por intermédio de 4.2, são de fato viáveis. Este procedimento será realizado ao adicionar, em 4.2, esferas centradas nos vértices das arestas



Figura 13 – Interseção de três esferas para o caso  $K = 3$ .



Fonte: (CAMARGO, 2015)

de poda (LAVOR; LIBERTI, 2014):

$$P = D \cap \left( \bigcap_{j=1}^{k_i} S(x_{v_j}, d_{v_j v_i}) \right), \quad (4.3)$$

onde  $k_i$  é a quantidade de arestas adicionais (de poda) associadas ao vértice  $v_i$  e as realizações contidas em 4.3 serão as classificadas como *válidas*.

Todavia, no caso onde  $k_i = 0$  para um dado vértice  $v_i$ , existirá a possibilidade, a medida que avança-se no espaço de busca, de seguir por caminhos que não atendam as restrições associadas a vértices posteriores a  $v_i$ . Isto fará com que seja necessário retroceder e tomar um outro caminho, a partir do nó escolhido erroneamente. Porém a quantidade de retrocessos pode crescer exponencialmente, dificultando novamente a resolução do problema (LAVOR *et al.*, 2017).

## 4.2 Simetria do $K$ DMDGP

Uma importante propriedade acerca do  $K$ DMDGP é a simetria das suas soluções. Caso não haja testes de viabilidade para um dado vértice  $v_i$ , tem-se que para qualquer solução viável, existirá uma outra solução simétrica a ela. Esta solução simétrica é obtida através do hiperplano gerado pelos  $K$  vértices anteriores ao vértice  $v_i$ . Tais hiperplanos são chamados de *planos de simetria* e podem ser obtidos antes da aplicação de qualquer algoritmo ou método.

O conjunto dos vértices que determinam os planos de simetria, denotado por  $Z$ , é definido da seguinte maneira (LAVOR *et al.*, 2017):

$$Z = \{v_t \in V : \nexists \{v_i, v_j\} \in E, \quad i + K < t \leq j\}. \quad (4.4)$$

Para exemplificar a construção do conjunto  $Z$ , consideremos o seguinte problema teste, retirado de (LAVOR *et al.*, 2017), para  $K = 3$ :

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\},$$

$$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_1, v_5\}, \{v_1, v_6\}, \\ \{v_2, v_3\}, \{v_2, v_4\}, \{v_2, v_5\}, \\ \{v_3, v_4\}, \{v_3, v_5\}, \{v_3, v_6\}, \\ \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_7\}, \\ \{v_5, v_6\}, \{v_5, v_7\}, \\ \{v_6, v_7\}\}$$

A matriz de conectividade associada ao problema, onde os círculos escuros representam as arestas pertencentes ao conjunto  $E$ , é dada por

.	1	2	3	4	5	6	7
1	●	●	●	●	●	●	○
2	○	●	●	●	●	○	○
3	○	○	●	●	●	●	○
4	○	○	○	●	●	●	●
5	○	○	○	○	●	●	●
6	○	○	○	○	○	●	●
7	○	○	○	○	○	○	●

O primeiro candidato a pertencer ao conjunto  $Z$  é  $v_4$ . Para que isto ocorra, não devem existir arestas  $\{v_i, v_j\}$  tal que  $i + 3 < 4 < j$ . De fato, não há vértice  $v_i$  cujo índice do terceiro posterior seja estritamente menor que 4, visto que o mesmo já é o de  $v_1$ , logo concluímos que  $v_4 \in Z$ . Isto significa que os vértices  $v_1, v_2$  e  $v_3$ , os três imediatamente anteriores, formam um plano de simetria, sendo este comum em qualquer  $KDMDGP$ .

Para  $v_5$ , não devem existir arestas  $\{i, j\}$  tal que  $i + 3 < 5 < j$ . Neste caso  $i = 1$  é possível pois  $v_4$ , seu terceiro posterior, possui índice menor que  $v_5$ , entretanto a aresta  $\{v_1, v_6\} \in E$ , logo  $v_5 \notin Z$ . Do mesmo modo, a aresta  $\{v_1, v_6\}$  faz com que o vértice  $v_6 \notin Z$ .

Para o vértice  $v_7$ , temos que  $v_4$  é exatamente o seu terceiro anterior, porém não existem arestas com vértices menores que o índice quatro, o que faz com que  $v_7 \in Z$ . Desta forma os vértices  $v_4, v_5$  e  $v_6$  também definem um plano de simetria, logo o conjunto  $Z$  é dado por  $\{v_4, v_7\}$ .

Tratemos então do modo como são utilizados os planos de simetria na determinação das soluções do  $KDMDGP$ . Para tal, utilizaremos sequências binárias para representar cada caminho na árvore binária, denotando os três primeiros vértices da clique inicial por 0, 0, 0. Seja então a primeira solução dada por

$$s_1 = (0, 0, 0, 0, 1, 0, 1).$$

o plano de simetria de  $v_7$  fornece a seguinte solução:

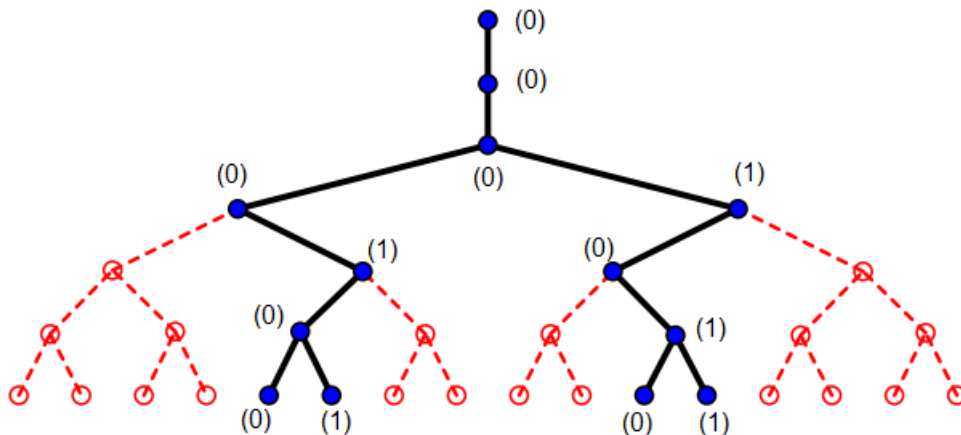
$$s_2 = (0, 0, 0, 0, 1, 0, 0),$$

enquanto que o plano de simetria de  $v_4$  fornece as soluções:

$$s_3 = (0, 0, 0, 1, 0, 1, 0) \quad \text{e} \quad s_4 = (0, 0, 0, 1, 0, 1, 1).$$

A Figura 14 apresenta estas quatro soluções obtidas, representadas por sequências binárias, na estrutura de árvore binária de busca.

Figura 14 – Soluções do exemplo anterior representadas na árvore binária de busca.



Fonte: (LAVOR; LIBERTI, 2014)

A quantidade de planos de simetria, ou seja, a cardinalidade do conjunto  $Z$ , permite fornecer o número de soluções do  $KDMDGP$ , que será  $2^{|Z|}$  (LAVOR *et al.*, 2017), visto que cada hiperplano gera duas opções. Este resultado permite afirmar que todas as soluções do  $KDMDGP$  podem ser geradas, utilizando apenas reflexões com respeito aos planos de simetria, a partir de qualquer uma obtida anteriormente (LAVOR *et al.*, 2017).

Tratando-se de complexidade, o  $KDMDGP$  é um problema NP-Difícil. Para classificar uma instância DGP como um  $KDMDGP$  é necessário encontrar uma ordenação dos seus vértices, e tal tarefa também é um problema NP-Difícil. No entanto, algumas características da

aplicação facilitam a obtenção de uma ordenação dos vértices, como por exemplo, a ordem dos átomos de uma proteína no cálculo de sua estrutura tridimensional (LAVOR *et al.*, 2012), sendo esta detalhada na seção seguinte.

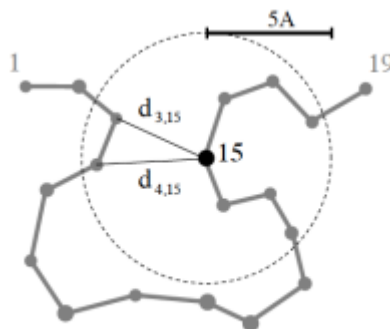
### 4.3 Ordenação nas Proteínas

Tanto esta seção quanto as seguintes passarão a abordar o caso tridimensional do  $K$ DMDGP, ou seja, o  $^3$ DMDGP. Embora tudo possa ser tratado no espaço  $K$ -dimensional, este caso particular foi escolhido por se tratar do enfoque principal deste trabalho: a conformação molecular de proteínas.

Existe uma ordenação natural entre os átomos da cadeia principal de uma proteína, sendo esta formada por uma sequência de três átomos ( $N$ ,  $C_\alpha$ ,  $C$ ), que repetem-se em toda a estrutura (LAVOR *et al.*, 2017).

O experimento de ressonância magnética nuclear fornece dados de átomos próximos em até  $5\text{\AA}$  (Figura 15) (CAMARGO, 2015). Dada então uma enumeração, temos que para os pares de átomos das posições  $i$  até  $i + 3$ , a distância é conhecida, pois são inferiores a  $5\text{\AA}$ . Isto significa que a RMN fornece a ordem desejada, pois há uma clique entre os átomos  $i$ ,  $i + 1$ ,  $i + 2$  e  $i + 3$ .

Figura 15 – Simulação 2D de um experimento de ressonância magnética nuclear.

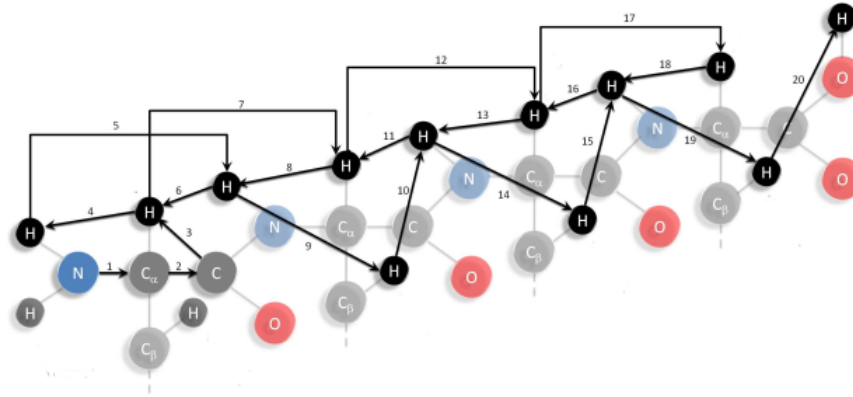


Fonte: (CAMARGO, 2015)

Todavia, a RMN é capaz de fornecer dados apenas entre os pares de hidrogênios e uma ordem entre estes não satisfaz as condições do  $K$ DMDGP. Para resolver este obstáculo, propõe-se um preenchimento com os átomos de hidrogênio das cadeias laterais, além de permitir repetições na ordem (Figura 16) (LAVOR *et al.*, 2017).

Esta repetição permite que as distâncias entre átomos de hidrogênio das posições  $i$  e

Figura 16 – Ordenação dos átomos de hidrogênio.



Fonte: (LAVOR *et al.*, 2017)

$i + 3$  sejam conhecidas, além de permitir a verificação e o controle de possíveis erros numéricos nos pontos já calculados (LAVOR *et al.*, 2017).

#### 4.4 Algoritmo *Branch-and-Prune*

O algoritmo *Branch-and-Prune* foi desenvolvido por (LIBERTI *et al.*, 2008), sendo o mais relevante algoritmo utilizado na resolução do  $KDMDGP$  com mais de 150 citações. Seu procedimento de resolução é através das chamadas matrizes de torção e numa busca em profundidade na árvore de coordenadas.

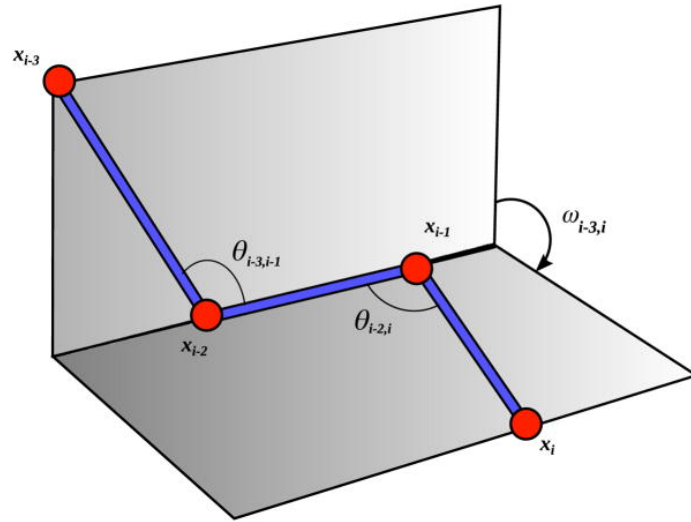
##### 4.4.1 Matrizes de Torção

Considere uma molécula com  $n$  átomos em sua cadeia principal, onde são conhecidas todas as distâncias  $d_{ij}$  entre pares de átomos covalentemente ligados, como também os ângulos planos  $\theta_{ij}$  formados por três átomos consecutivos (PHILLIPS *et al.*, 1994). Além disto, são conhecidos os senos e cossenos dos ângulos de torção  $\omega_{ij}$ , correspondentes a quatro átomos consecutivos (Figura 17).

Para quatro átomos consecutivos  $x_i, x_{i-1}, x_{i-2}$  e  $x_{i-3}$ , a expressão do cosseno do ângulo de torção  $\omega_{i-3,i}$  em função das distâncias  $d_{i-1,i}, d_{i-2,i}, d_{i-3,i-1}$  e dos ângulos planos  $\theta_{i-2,i}$  e  $\theta_{i-3,i-1}$  é dada por (LIBERTI *et al.*, 2008):

$$\cos\omega_{i-3,i} = \frac{d_{i-1,i}^2 + d_{i-3,i-1}^2 - 2 \cdot d_{i-1,i} \cdot d_{i-3,i-1} \cdot \cos\theta_{i-2,i} \cdot \cos\theta_{i-3,i-1} - d_{i-2,i}^2}{2 \cdot d_{i-1,i} \cdot d_{i-3,i-1} \cdot \sin\theta_{i-2,i} \cdot \sin\theta_{i-3,i-1}}, \quad (4.5)$$

Figura 17 – Distâncias intermoleculares e ângulos de torção.



Fonte: (LAVOR *et al.*, 2019)

onde os senos dos ângulos planos  $\theta_{ij}$  são facilmente obtidos pela relação fundamental trigonométrica.

No algoritmo *Branch-and-Prune* todas as coordenadas cartesianas  $x_1, x_2, \dots, x_n \in \mathbb{R}^3$  dos átomos podem ser determinadas pela seguinte fórmula (PHILLIPS *et al.*, 1994):

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ 1 \end{bmatrix} = B_1 \cdot B_2 \cdot \dots \cdot B_n \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad \forall i = 1, \dots, n \quad (4.6)$$

onde:

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1 & 0 & 0 & -d_{1,2} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$B_3 = \begin{bmatrix} -\cos\theta_{1,3} & -\sin\theta_{1,3} & 0 & -d_{2,3} \cdot \cos\theta_{1,3} \\ \sin\theta_{1,3} & -\cos\theta_{1,3} & 0 & d_{2,3} \cdot \sin\theta_{1,3} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$B_i = \begin{bmatrix} -\cos\theta_{i-2,i} & -\sin\theta_{i-2,i} & 0 & -d_{i-1,i} \cdot \cos\theta_{i-2,i} \\ \sin\theta_{i-2,i} \cdot \cos\omega_{i-3,i} & -\cos\theta_{i-2,i} \cdot \cos\omega_{i-3,i} & -\sin\omega_{i-3,i} & d_{i-1,i} \cdot \sin\theta_{i-2,i} \cdot \cos\omega_{i-3,i} \\ \sin\theta_{i-2,i} \cdot \sin\omega_{i-3,i} & -\cos\theta_{i-2,i} \cdot \sin\omega_{i-3,i} & \cos\omega_{i-3,i} & d_{i-1,i} \cdot \sin\theta_{i-2,i} \cdot \sin\omega_{i-3,i} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.7)$$

onde  $B_i$  é chamada de *matriz de torção* do átomo da posição  $i$  e  $C_i = \prod_{i=1}^n B_i$  é a chamada *matriz de torção acumulada* do átomo da mesma posição.

Uma das condições do  $^3DMDGP$  é a existência de uma realização válida para, pelo menos, os três primeiros átomos. No *Branch-and-Prune* estes átomos são fixados nas seguintes posições:

$$x_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$x_2 = \begin{bmatrix} -d_{1,2} \\ 0 \\ 0 \end{bmatrix},$$

$$x_3 = \begin{bmatrix} -d_{1,2} + d_{2,3} \cdot \cos\theta_{1,3} \\ d_{2,3} \cdot \sin\theta_{1,3} \\ 0 \end{bmatrix}$$

A partir do quarto átomo tem-se duas posições possíveis, cada uma proveniente do seno e cosseno do ângulo de torção. São geradas duas matrizes de torção  $B_4$  e  $B'_4$ , cada qual responsável por uma diferente posição. As prováveis coordenadas do quarto átomo são dadas por (LIBERTI *et al.*, 2008):

$$x_4 = \begin{bmatrix} -d_{1,2} + d_{2,3} \cdot \cos\theta_{1,3} - d_{3,4} \cdot \cos\theta_{1,3} \cdot \cos\theta_{2,4} + d_{3,4} \cdot \sin\theta_{1,3} \cdot \sin\theta_{2,4} \cdot \cos\omega_{1,4} \\ d_{2,3} \cdot \sin\theta_{1,3} - d_{3,4} \cdot \sin\theta_{1,3} \cdot \cos\theta_{2,4} - d_{3,4} \cdot \cos\theta_{1,3} \cdot \sin\theta_{2,4} \cdot \cos\omega_{1,4} \\ -d_{3,4} \cdot \sin\theta_{2,4} \cdot \sqrt{1 - (\cos\omega_{1,4})^2} \end{bmatrix},$$

$$x'_4 = \begin{bmatrix} -d_{1,2} + d_{2,3} \cdot \cos\theta_{1,3} - d_{3,4} \cdot \cos\theta_{1,3} \cdot \cos\theta_{2,4} + d_{3,4} \cdot \sin\theta_{1,3} \cdot \sin\theta_{2,4} \cdot \cos\omega_{1,4} \\ d_{2,3} \cdot \sin\theta_{1,3} - d_{3,4} \cdot \sin\theta_{1,3} \cdot \cos\theta_{2,4} - d_{3,4} \cdot \cos\theta_{1,3} \cdot \sin\theta_{2,4} \cdot \cos\omega_{1,4} \\ d_{3,4} \cdot \sin\theta_{2,4} \cdot \sqrt{1 - (\cos\omega_{1,4})^2} \end{bmatrix}.$$

Para o quinto átomo haverá quatro possibilidades, pois novamente são criadas duas matrizes de torção ( $B_5$  e  $B'_5$ ) que, combinados às referentes ao átomo anterior, geram:

$$B_4 \cdot B_5; B_4 \cdot B'_5; B'_4 \cdot B_5 \text{ e } B'_4 \cdot B'_5$$

Este caráter combinatório permite afirmar que, para o átomo da posição  $i$  da cadeia principal com  $i \geq 4$ , temos  $2^{i-3}$  possibilidades de conformações distintas. Este resultado viabiliza a discretização do problema e permite trabalhá-lo como uma árvore de busca.

O algoritmo *Branch-and-Prune* então realiza uma busca em profundidade e ramificada sobre todas as posições que são viáveis com respeito às restrições, enquanto que as inviáveis são descartadas (SOUZA, 2010). Este procedimento é visto mais detalhadamente na estrutura e no pseudo-código do *Branch-and-Prune*.

#### 4.4.2 Estrutura do *Branch-and-Prune*

Seja  $T$  a árvore de busca construída a partir das possibilidades de posições para cada átomo da cadeia principal da molécula. Os três primeiros átomos já possuem posições determinadas conforme apresentado e a cada etapa que se segue temos quatro possibilidades para o átomo da posição  $i$ :

1. Se  $x_i$  e  $x'_i$  são viáveis, então ambas as posições são armazenadas e exploradas mais profundamente em  $T$ ;
2. Se  $x_i$  é viável e  $x'_i$  inviável, então  $x_i$  é armazenado enquanto  $x'_i$  é descartado;
3. Se  $x'_i$  é viável e  $x_i$  inviável, então  $x'_i$  é armazenado enquanto  $x_i$  é descartado;
4. Se  $x_i$  e  $x'_i$  são inviáveis, então todo o ramo é descartado e retrocede-se para o nível anterior em  $T$ .

Para cada nó do nível  $i$  na árvore de busca  $T$  são armazenadas quatro informações:

- A posição  $x_i$  do  $i$ -ésimo átomo;
- A matriz de torção acumulada  $C_i$  associada ao  $i$ -ésimo átomo;
- Um ponteiro  $P(i)$  para o nó pai do vértice  $i$ ;
- Ponteiros para os sub-nós filhos  $L(i)$  e  $R(i)$ .

O pseudo-código do algoritmo *Branch-and-Prune* presente em (LIBERTI *et al.*, 2008), onde  $E_P$  consiste no conjunto das arestas de poda e  $\varepsilon$  uma dada tolerância, é detalhado a seguir:



**Algorithm 1 Branch-and-Prune (BP)****BP** ( $T, v, i$ );**if** ( $i \leq n - 1$ ) **then****CALCULO DAS POSSÍVEIS POSIÇÕES PARA O  $i$ -ÉSIMO ÁTOMO;**Calcular as matrizes de torção  $B_i$  e  $B'_i$  através de 4.7;Acessar a matriz de torção acumulada  $C_{i-1}$ , a partir do nó pai  $P(v)$ ;Calcular  $C_i \leftarrow C_{i-1} \cdot B_i$  e  $C'_i \leftarrow C_{i-1} \cdot B'_i$ ;Calcular  $x_i \leftarrow C_i \cdot y$  e  $x'_i \leftarrow C'_i \cdot y$ , onde  $y = (0, 0, 0, 1)^T$ ;Seja  $\lambda \leftarrow 1$  e  $\rho \leftarrow 1$ ;**TESTE DE VIABILIDADE;****for** ( $(i, j) \in E_P$ ) **do**Seja  $\delta_{ij} \leftarrow (\|x_i - x_j\|^2 - d_{ij}^2)^2$  e  $\delta'_{ij} \leftarrow (\|x'_i - x_j\|^2 - d_{ij}^2)^2$ ;**if** ( $\delta_{ij} > \varepsilon$ ) **then**|  $\lambda \leftarrow 0$ **end****if** ( $\delta'_{ij} > \varepsilon$ ) **then**|  $\rho \leftarrow 0$ **end****end****CRIAÇÃO DOS SUB-NÓS;****if** ( $\lambda = 1$ ) **then**Criar nó  $z$ ;Armazenar matriz de torção acumulada  $C_i$  e  $x_i$  em  $z$ ; $P(z) \leftarrow v$ ; $L(v) \leftarrow z$ ; $T \leftarrow T \cup \{z\}$ ;**BP** ( $T, z, i + 1$ );**else** $L(v) \leftarrow \text{PODADO}$ ;**end****if** ( $\rho = 1$ ) **then**Criar nó  $z'$ ;Armazenar matriz de torção acumulada  $C_i$  e  $x'_i$  em  $z'$ ; $P(z') \leftarrow v$ ; $R(v) \leftarrow z'$ ; $T \leftarrow T \cup \{z'\}$ ;**BP** ( $T, z', i + 1$ );**else** $R(v) \leftarrow \text{PODADO}$ ;**end****else** $\text{SOLUÇÃO ENCONTRADA}$ ;**end**

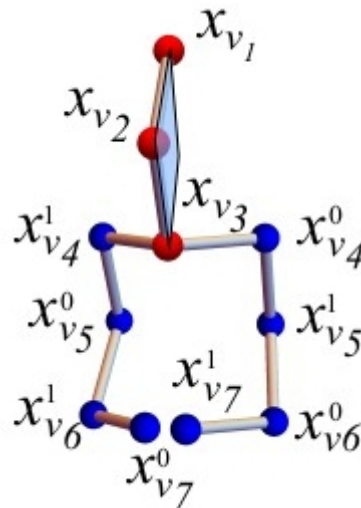
## 5 METODOLOGIA

Neste capítulo abordaremos a idéia proposta para este trabalho, apresentando os obstáculos que surgiram nos métodos já desenvolvidos e que motivaram este estudo, assim como as estratégias utilizadas para resolvê-los.

Como foi citado, o  $KDMDGP$  (*K-Problema Discretizável de Geometria de Distâncias Moleculares*) consiste em uma abordagem discreta do DGP aplicada às moléculas de proteínas. Esta abordagem trata o problema como uma árvore binária de busca, construída e explorada pelo algoritmo *Branch-and-Prune* (*BP*), isto a partir de uma ordenação dos vértices (átomos), previamente exigida na definição do problema.

As soluções do  $KDMDGP$  são representadas por sequências binárias, que no contexto dos grafos, consistem em caminhos do nó raiz até uma das suas folhas. Dentre as propriedades presentes no  $KDMDGP$ , uma de particular importância é a simetria das suas soluções (Figura 18), pois através dela é possível, em uma etapa pós-solução, determinar todas as soluções a partir de alguma fornecida pelo *BP* ou por métodos de abordagem contínua. No entanto, é necessário que sejam conhecidos todos os planos de simetria da estrutura tridimensional da molécula (LAVOR *et al.*, ).

Figura 18 – Soluções simétricas.



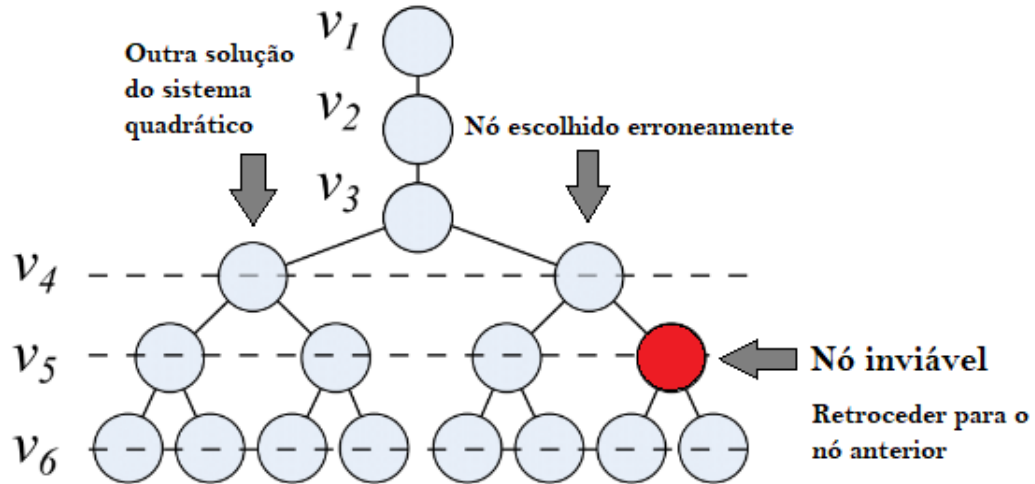
Fonte: (LAVOR; LIBERTI, 2014)

Apesar desta vantagem, o algoritmo *Branch-and-Prune* pode vir a realizar os mesmos cálculos repetidas vezes em sua execução. Esta situação ocorrerá caso as posições dos átomos tenham sido escolhidas erroneamente por causa da falta de testes de viabilidade, conforme Figura 19.

Tais testes são os responsáveis por garantir que as posições obtidas por intermédio dos sistemas quadráticos sejam viáveis em relação às arestas adicionais e suas respectivas equações. Quando tais erros ocorrem, faz-se necessário retroceder o caminho na árvore binária até o vértice que gerou a

inviabilidade e tomar outra opção de posição para o átomo, no mesmo nível da árvore, retomando assim o caminho e os cálculos a partir desta nova posição. Exemplos acerca do procedimento de obtenção das posições de cada átomo vide (LAVOR; LIBERTI, 2014).

Figura 19 – Esquemática do retrocesso de posições.



Fonte: (LAVOR; LIBERTI, 2014)

Uma quantidade excessiva de retrocessos faz com que o algoritmo *Branch-and-Prune* desperdice muito tempo computacional ao ter que refazer o caminho correspondente a uma solução do  $K$ DMDGP. Todavia, a simetria das soluções permite evitar o número exagerado (eventualmente exponencial) de repetições de cálculos que possam surgir durante a execução do algoritmo.

A redução da quantidade de operações realizadas será a base da nova versão do algoritmo BP, chamada de *Symmetric Branch-and-Prune (SBP)*, cuja ideia é apresentada na próxima seção.

## 5.1 Symmetric Branch-and-Prune

Nesta seção, apresentaremos o suporte teórico da versão simétrica do algoritmo *Branch-and-Prune*. Para tal, precisamos inicialmente enunciar um teorema de suma importância para o entendimento da ideia por trás desta nova abordagem. Esse teorema envolve conjuntos e operadores que não foram definidos anteriormente. Contudo, utilizaremos algumas ilustrações da árvore de coordenadas para facilitar a compreensão do resultado proposto.

**Teorema 5.1.1** ((LIBERTI *et al.*, 2014)). *Com probabilidade 1, para todo  $v > K$  e  $u < v - K$  existe um conjunto  $H^{uv}$  de  $2^{v-u-K}$  valores reais positivos tal que, para cada realização  $x \in X$ , temos que  $\|x_v - x_u\| \in H^{uv}$ . Além disso,  $\forall x' \in X$ ,  $\|x_v - x_u\| = \|x'_v - x_u\|$  se, e somente se,  $x'_v \in \{x_v, R_x^{u+K}(x_v)\}$ , sendo  $R_x^v$  o operador de reflexão.*

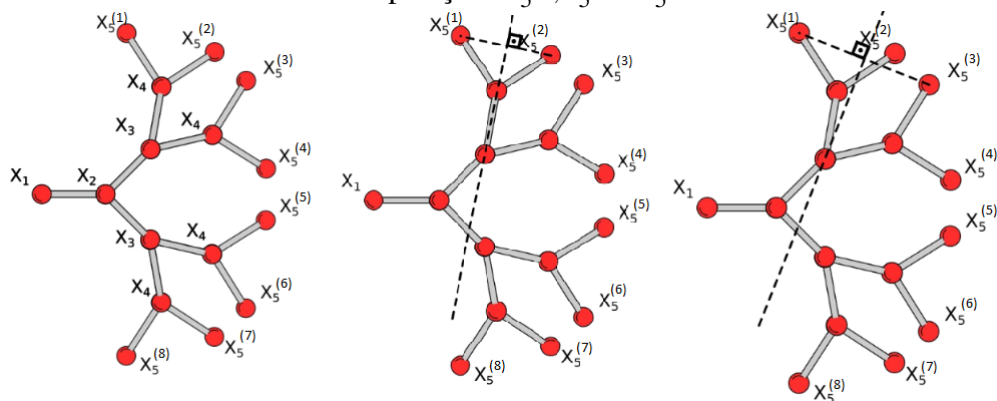
O teorema acima diz que, com probabilidade 1, o conjunto  $H^{uv}$  das distâncias  $\|x_v - x_u\|$  para quaisquer realizações dos vértices  $u$  e  $v$  possui cardinalidade  $2^{v-u-K}$ . Por exemplo, para  $u = 1$ ,  $v = 5$  e  $K = 2$  (caso bidimensional), teremos quatro possibilidades de realização para  $x_5$  sendo elas:  $(x_5^{(1)}, x_5^{(2)}, x_5^{(3)}, x_5^{(4)})$ . Entretanto, de acordo com a árvore de coordenadas apresentada na Figura 20, existem outras quatro possibilidades para o vértice  $x_5$ :  $(x_5^{(5)}, x_5^{(6)}, x_5^{(7)}, x_5^{(8)})$ . Isto não contradiz o teorema acima, pois de acordo com a geometria da estrutura, fato este ainda a ser detalhado, temos que dada uma realização existirá outra que é a sua reflexão com respeito ao vértice de interesse. Para o exemplo totalizamos 8 possibilidades para a posição do vértice  $x_5$ , porém com quatro distâncias possíveis.

Dando continuidade ao teorema 5.1.1, este ainda afirmará que se existir, de acordo com o exemplo apresentado, uma aresta de poda conectando vértices  $x_1$  e  $x_5$ , então só existirão duas realizações válidas para o vértice  $x_5$  sendo uma a reflexão da outra obtida através do operador de reflexão  $R_x^v$ . Constatemos então a existência e a unicidade desta outra realização válida através de argumentos geométricos, no entanto é necessário deixar claro que os argumentos que serão apresentados unicamente tem o objetivo de ilustrar o resultado do teorema, não fazendo parte nem do processo de construção da árvore de coordenadas e nem do cálculo da solução do problema.

Vale também mencionar que a argumentação geométrica a ser apresentada não é uma demonstração do teorema, mas apenas uma aplicação. A prova do teorema encontra-se em (LIBERTI *et al.*, 2014) e a argumentação tem finalidade apenas ilustrativa.

Dada a árvore de coordenadas (Figura 20) do exemplo sugerido anteriormente ( $u = 1$ ,  $v = 5$  e  $K = 2$ ), visualizam-se as já mencionadas 8 possíveis posições para o vértice  $x_5$ , além disso suponhamos que a primeira delas  $x_5^{(1)}$  seja a realização válida, isto é,  $\|x_1 - x_5^{(1)}\| = d_{1,5}$ . Entretanto, ao construir-se a mediatriz entre  $x_5^{(1)}$  e  $x_5^{(2)}$  verifica-se que o vértice  $x_1$  não pertence a mediatriz, logo a posição  $x_5^{(2)}$  não será válida, pois esta não satisfaz a restrição  $\|x_1 - x_5^{(2)}\| = d_{1,5}$ .

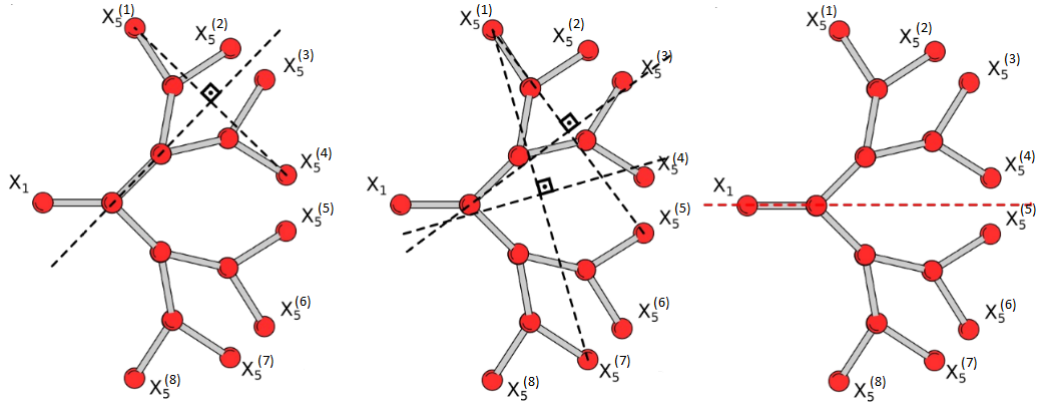
Figura 20 – Mediatrizes referentes as posições  $x_5^{(1)}, x_5^{(2)}$  e  $x_5^{(3)}$ .



O mesmo acontece com cada uma das mediatrizes construídas entre os demais pares de

posições, exceto para aquela entre a primeira ( $x_5^{(1)}$ ) e a última ( $x_5^{(8)}$ ) das posições para  $x_5$ . Essa mediatriz é o plano (reta) de simetria definido por  $x_1$  e  $x_2$  vide Figura 21.

Figura 21 – Mediatrizes referentes as posições  $x_5^{(1)}, x_5^{(4)}, x_5^{(5)}, x_5^{(7)}$  e  $x_5^{(8)}$ .



Temos então que a primeira ( $x_5^{(1)}$ ) e a última ( $x_5^{(8)}$ ) das possíveis posições para  $x_5$  distam igualmente ( $d_{1,5}$ ) de  $x_1$ . Logo ambas satisfazem a aresta  $\{1, 5\}$ . Além disso, como a mediatriz é o plano de simetria, essa última posição ( $x_5^{(8)}$ ) para  $x_5$  consiste exatamente na representação simétrica da primeira ( $x_5^{(1)}$ ), confirmando assim o que é proposto no teorema apresentado inicialmente nesta seção.

Após a explicação do teorema 5.1.1 e a título de uma visualização prática, será apresentado um exemplo no espaço bidimensional, uma vez que os demais consistem em generalizações do que será exposto.

Considere o conjunto de vértices  $V = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$  e o conjunto de arestas

$$E = \{\{1, 2\}, \{1, 3\}, \{1, 5\}, \{1, 7\}, \{1, 9\}, \{2, 3\}, \{2, 4\}, \{2, 6\}, \{2, 8\}, \{3, 4\}, \{3, 5\}, \{3, 7\}, \{3, 9\}, \{4, 5\}, \{4, 6\}, \{4, 8\}, \{5, 6\}, \{5, 7\}, \{5, 9\}, \{6, 7\}, \{6, 8\}, \{7, 8\}, \{7, 9\}, \{8, 9\}\}.$$

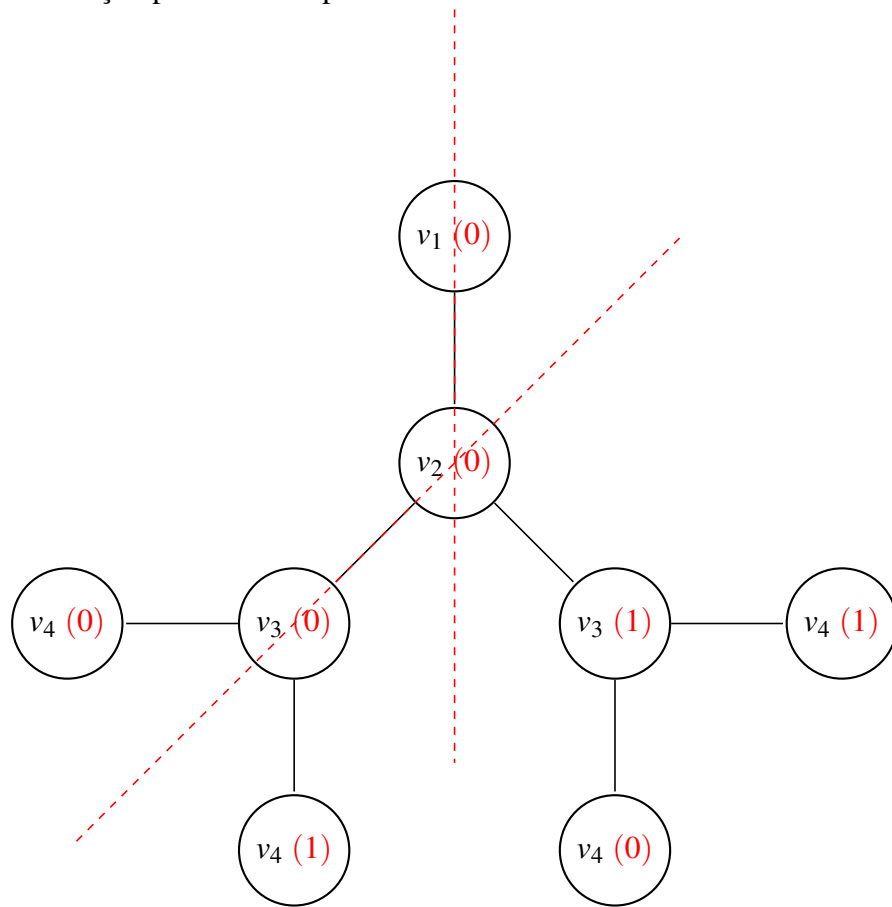
A matriz de conectividade de  $G = (V, E)$  pode ser representada pela Figura 22.

.	1	2	3	4	5	6	7	8	9
1	●	●	●	○	●	○	●	○	●
2	○	●	●	●	○	●	○	●	○
3	○	○	●	●	●	○	●	○	●
4	○	○	○	●	●	●	○	●	○
5	○	○	○	○	●	●	●	○	●
6	○	○	○	○	○	●	●	●	○
7	○	○	○	○	○	○	●	●	●
8	○	○	○	○	○	○	○	●	●
9	○	○	○	○	○	○	○	○	●

Figura 22 – Matriz de conectividade do caso bidimensional.

Em duas dimensões, para calcular as possíveis posições de um dado vértice, são necessárias apenas as informações sobre os dois vértices imediatamente anteriores, onde cada par consecutivo define uma reta que divide o espaço bidimensional. Tomando a regra da mão-direita, definimos uma orientação para os semi-planos, onde os vértices posicionados abaixo (esquerda) da reta serão marcados com o valor “0” e os acima (direita) com o valor “1”. Mesmo não possuindo dois antecessores, arbitramos o valor “0” aos dois primeiros vértices (Vide Figura 23):

Figura 23 – Orientação para os semi-planos.



Cada combinação de posições a ser testada consiste em uma sequência binária:

$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	...	$b_9$
0	0	0	0	0	...	0
0	0	0	0	0	...	1
0	0	0	0	1	...	0
0	0	0	1	1	...	0
...						
0	0	1	1	1	...	0
0	0	1	1	1	...	1

Consideremos então que as posições calculadas para os vértices 3, 4 e 5 associadas à representação  $(b_3, b_4, b_5) = (0, 1, 0)$  satisfaçam a aresta  $\{1, 5\}$ . Entretanto, através do teorema 5.1.1 que foi apresentado anteriormente, pode-se afirmar a existência de uma outra representação binária que também satisfaz a aresta  $\{1, 5\}$ .

Ao aplicarmos então este resultado no exemplo proposto anteriormente temos que, se a representação  $(b_3, b_4, b_5) = (0, 1, 0)$  satisfaz a aresta  $\{1, 5\}$ , então apenas a solução associada à representação simétrica  $(b_3, b_4, b_5) = (1, 0, 1)$  também a satisfaz.

Segue-se então que esta nova abordagem do algoritmo *Branch-and-Prune* armazenará o trecho  $(b_3, b_4, b_5) = (1, 0, 1)$  evitando que seja desfeito e tenha de ser recalculado. Deste modo, as subestruturas corretas já determinadas são utilizadas como blocos fixos. Um resultado imediato desta proposta, em conjunto com o Teorema 5.1.1, é a redução na quantidade de operações realizadas, pois um trecho já calculado e armazenado terá apenas duas possibilidades, sendo uma a simétrica da outra.

Isto pode ser ilustrado na busca por uma solução viável para a aresta  $\{1, 7\}$  do exemplo apresentado. Supondo que tenhamos obtido uma subestrutura válida para a aresta  $\{1, 5\}$ , já fixamos e armazenamos a representação  $(b_3, b_4, b_5) = (0, 1, 0)$ . Portanto, resta determinar apenas  $b_6$  e  $b_7$ . Sendo assim, temos oito possibilidades:

$b_3$	$b_4$	$b_5$	$b_6$	$b_7$
0	1	0	0	0
0	1	0	0	1
0	1	0	1	0
0	1	0	1	1
1	0	1	0	0
1	0	1	0	1
1	0	1	1	0
1	0	1	1	1

O total de combinações decresce de  $2^5$  para  $2^3$  e, caso a sequência  $(0, 1, 0)$  não seja uma solução, a sua simétrica  $(1, 0, 1)$  facilmente obtida o será. Quanto maiores forem as quantidades de blocos já calculados em um trecho, maior será a redução na quantidade de operações realizadas.

É importante mencionar que mesmo partes de um bloco viável, ou seja, um bloco que represente coordenadas que satisfazem uma restrição, podem ser úteis na construção de novos blocos. Por exemplo, na busca por soluções para a aresta  $\{2, 8\}$ , os vértices 2 e 3 formam o eixo e deve-se determinar as posições para os vértices 4 ao 8. Ora, os vértices 4 e 5 fazem parte de um bloco já calculado, portanto são acoplados, reduzindo assim da mesma forma o número de cálculos.

Outra importante vantagem desta ideia baseia-se no princípio de “*dividir-para-conquistar*”, ou seja, a resolução de pequenos problemas facilita a resolução de trechos maiores da estrutura. Tudo isto é evidenciado, por exemplo, no cálculo da aresta  $\{1, 7\}$ , pois, se as arestas  $\{1, 5\}$  e  $\{3, 7\}$  já tiverem sido ajustadas, então todas as posições dos vértices 3 ao 7 já foram determinadas e estão armazenadas. Isto permite que o algoritmo não precise dispendir esforço em resolver a aresta  $\{1, 7\}$ .

A aresta  $\{i, j\}$  é satisfeita se a distância entre os vértices  $i$  e  $j$  for correta. Como consequência, se soubermos associar cada representação binária diretamente às coordenadas dos vértices  $i$  e  $j$ , podemos primeiro identificar a corretude da representação aplicando-a exclusivamente aos vértices  $i$  e  $j$  e, depois, utilizá-la para fixar os demais vértices. Esta associação independente da representação binária às coordenadas  $x_i$  e  $x_j$  pode ser obtida via operadores de reflexão. Este procedimento transformará a busca na árvore de coordenadas em uma busca na árvore de reflexões. A seguir, será apresentado o detalhamento teórico sobre o operador de reflexão:

## 5.2 Reflexão geométrica

Em matemática, uma *isometria* (*iso* = igual, *metria* = medida) é uma transformação que mapeia os elementos de um espaço métrico em outro ou nele mesmo, tal que as distâncias originais sejam preservadas entre os elementos da imagem no novo espaço. Existem três tipos de isometrias: translação, rotação e reflexão.

A reflexão, isometria de interesse deste trabalho, ocorre em relação a um hiperplano que, no  $\mathbb{R}^2$ , é chamado de *eixo de reflexão* e, no  $\mathbb{R}^3$ , de *plano de reflexão*. Nela, um dado ponto e o seu correspondente refletido estão sobre uma reta perpendicular ao hiperplano de reflexão e mesma distância deste.

Uma reflexão  $f$  é uma *involução*, isto é, se aplicada duas vezes seguidas, o ponto retornará a sua posição original e, conseqüentemente, todo objeto geométrico é restaurado ao seu estado original. Esta propriedade é representada pela expressão:

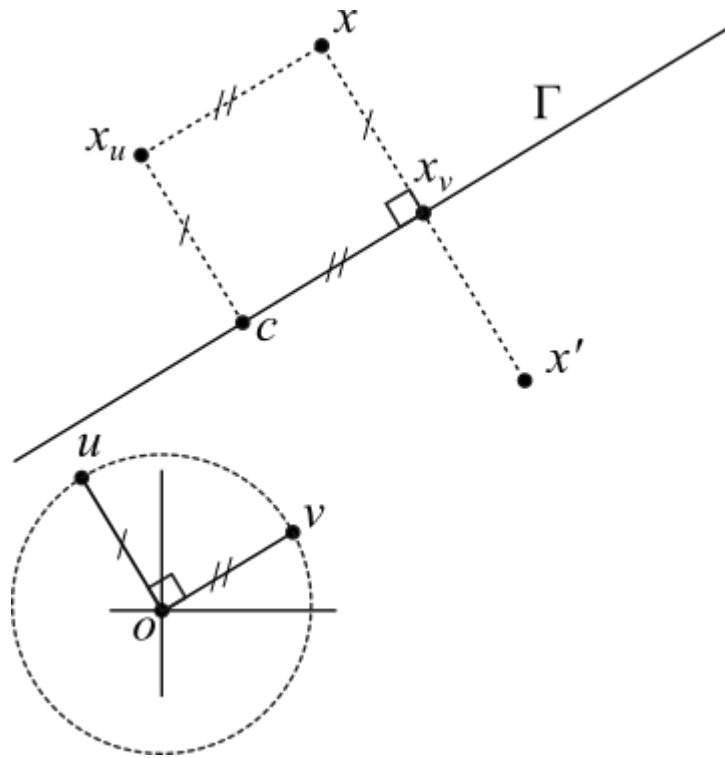
$$f(f(x)) = x. \quad (5.1)$$

Fixando a representação canônica dos espaços vetoriais  $\mathbb{R}^n$  como matrizes reais  $n \times 1$ , usaremos a notação  $u^t v$  para representar o produto interno canônico entre  $\vec{u}$  e  $\vec{v}$  em  $\mathbb{R}^n$ . Além disso, não faremos distinção entre vetores e pontos.

Apresentemos então o cálculo para obter a reflexão  $x'$  de um ponto  $x$ . O espaço geométrico da Figura 24 é o bidimensional, porém, a expressão obtida é válida para o espaço  $n$ -dimensional.



Figura 24 – Reflexão Bidimensional.



Considere o hiperplano de reflexão  $\Gamma$  com vetor normal unitário  $u$  e que contém o ponto  $c$  cuja equação é dada por

$$u^t(x - c) = 0 \quad (5.2)$$

Tomando  $x$ , um ponto qualquer no espaço, e um vetor  $v$  unitário e perpendicular a  $u$ , podemos escrever

$$x = c + x_v + x_u,$$

com  $x_v$  e  $x_u$  sendo as projeções ortogonais de  $x$  sobre os vetores  $u$  e  $v$ . Sabendo que  $x_u = c + u^t(x - c)u$ , a reflexão de  $x$  com respeito ao plano  $\Gamma$  é dada por

$$x' = c + x_v - x_u, \quad (5.3)$$

$$= c + (x - c - x_u) - (c + u^t(x - c)u) \quad (5.4)$$

$$= x - c - x_u - u^t(x - c)u \quad (5.5)$$

$$= -x_u + (I - uu^t)(x - c) \quad (5.6)$$

$$= -(c + u^t(x - c)u) + (I - uu^t)(x - c) \quad (5.7)$$

$$= (I - 2uu^t)(x - c) - c, \quad (5.8)$$

onde 5.8 é a equação de reflexão de  $x$ .

Para o  $KDMDGP$ , serão aplicadas reflexões tridimensionais e os hiperplanos são os planos de simetria da estrutura molecular de proteínas.

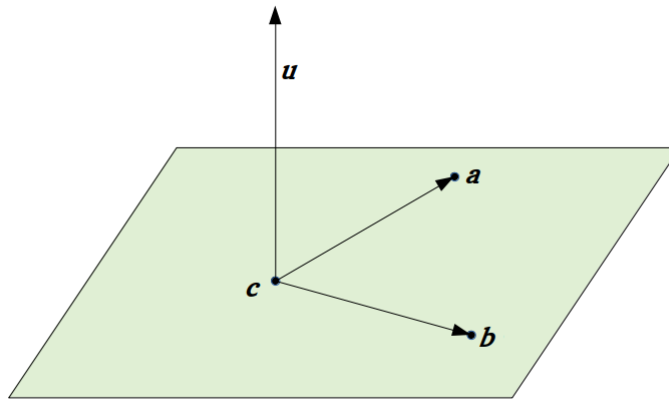
Considere então  $a, b$  e  $c \in \mathbb{R}^3$ , três vértices que geram um plano de simetria  $\Gamma$  (Figura 25). Um vetor  $u$  normal unitário ao plano  $\Gamma$  pode ser obtido a partir dos pontos  $a, b$  e  $c$ , tomando

$$u = \frac{w \times r}{\|w \times r\|} \quad (5.9)$$

onde  $w = (a - c)$ ,  $r = (b - c)$  e  $w \times r$  é o produto vetorial definido por

$$w \times r = \begin{vmatrix} i & j & k \\ w_1 & w_2 & w_3 \\ r_1 & r_2 & r_3 \end{vmatrix}. \quad (5.10)$$

Figura 25 – Produto Vetorial.

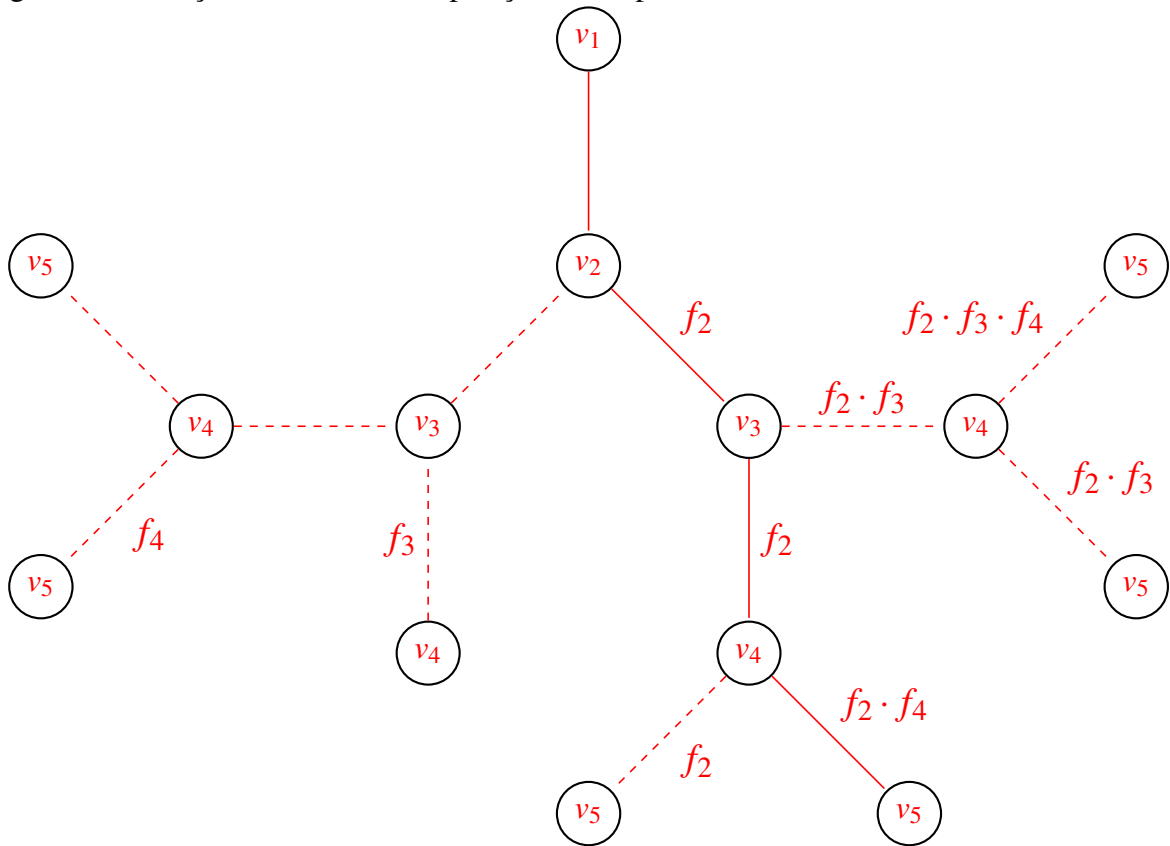


Apresentado o embasamento teórico acerca do operador de reflexão, passemos então para sua aplicação na construção da árvore de reflexões que permitirá obter as posições relativas para cada vértice.

Cada posição relativa de um vértice  $j$  é obtida pela aplicação de uma sequência (composição) de reflexões, que são construídas a partir dos hiperplanos de reflexão  $\Gamma$  conforme a equação 5.8. Chamemos estas aplicações de  $f_j$ , com  $j \in \{2, 3, \dots, n\}$  com  $n$  sendo o número de hiperplanos de reflexão.

Uma reflexão  $f_2$  fornecerá duas posições relativas, uma por sua aplicação e outra pela não aplicação. A presença de duas reflexões  $f_2$  e  $f_3$ , fornece 4 combinações de posições: não aplicar nenhuma, apenas  $f_2$ , apenas  $f_3$  ou aplicar ambas. Logo, para  $n$  reflexões tem-se acesso as  $2^{i-k}$  posições relativas para um dado vértice  $i$ , conforme a Figura 26.

Figura 26 – Posições relativas e composições dos operadores de reflexão.



Todas as possíveis posições para um vértice  $i$  são obtidas pela sequência (composição) dos operadores de reflexão  $f_j$  para  $j \in \{K, \dots, i-1\}$ . Além disto, descoberta a posição que satisfaça uma restrição associada à uma dada aresta  $\{i, j\}$ , aplica-se a mesma sequência de reflexões nos demais vértices  $j > i$ .

É importante mencionar que nem todas as possibilidades são válidas, pois elas devem preservar os trechos já calculados e armazenados. Por exemplo, se o trecho  $(b_3, b_4, b_5) = (1, 0, 1)$ , então uma possibilidade que utilize  $b_4$  e  $b_5$  com valor 1 é infactível, conforme o que foi apresentado no Teorema 5.1.1.

O Algoritmo 2 incorpora as ideias centrais do método *Symmetric Branch-and-Prune* (SBP). porém ressaltamos que uma das condições prévias para sua inicialização, as demais seguem da própria definição do  $K$ DMDGP, é a de que os  $K$  primeiros vértices já estejam fixados.

Neste algoritmo também definimos o conjunto  $B(\{i+K, \dots, j-1\})$  que é responsável pelo armazenamento das subestruturas válidas já calculadas entre os vértices de índices  $i+K$  e  $j-1$ , com  $K$  sendo a dimensão do espaço. É importante mencionar que no início do algoritmo, cada subestrutura é composta por apenas um vértice e, a cada aresta  $\{i, j\}$  ajustada, as subestruturas menores são substituídas pela que foi recém calculada, no algoritmo chamamos este procedimento de *ATUALIZAR* o conjunto  $B$ .

A aplicação da sequência de operadores (composição) de reflexão  $f_k$  é definida através da

seguinte função:

$$g_w(x) = f_{i_1} \cdot f_{i_2} \cdot \dots \cdot f_{i_t}(x) \quad (5.11)$$

onde  $i_1 < i_2 < \dots < i_t$  e  $\{i_1, i_2, \dots, i_t\} \subset B$

---

**Algorithm 2** *Symmetric Branch-and-Prune (SBP)*

---

$V$ : Conjunto de Vértices;

$E_p$ : Conjunto de arestas de poda;

$d$ : Função de distância, onde  $d_{ij}$  representa a distância entre vértices  $x_i$  e  $x_j$ ;

$x_j^0$ : Configuração inicial de referência do vértice  $j$ ;

**for**  $\{i, j\} \in E_p$  **do**

    Calcular hiperplanos de reflexão  $\Gamma_k$ ,  $\forall k \in B(\{i + K, \dots, j - 1\})$  (Equação 5.2);

    Construir operadores de reflexão  $f_k$  a partir dos hiperplanos  $\Gamma_k$  (Equação 5.8);

    Calcular  $g_w(x_j)$  para cada sequência de  $f_k$ 's, com  $k \in B(\{i + K, \dots, j - 1\})$  a partir de  $x_j^0$  (Figura 26);

**if**  $\|x_i - x_j\| = d_{ij}$  **then**

        Calcular  $g_w(x_k)$ , para  $k \in B(\{i + K, \dots, j - 1\})$ ;

**ATUALIZAR**  $B(\{i + K, \dots, j - 1\})$

**end**

**end**

---

## 6 RESULTADOS

Neste capítulo, serão apresentados os detalhes dos experimentos computacionais realizados e resultados obtidos. Estes serão apresentados nas tabelas presentes no final deste capítulo.

Os experimentos numéricos realizados comparam os resultados obtidos pelas implementações MDJEEP (Mucherino) e *Symmetric Branch-and-Prune* (SBP). O MDJEEP consiste em uma implementação do próprio criador do algoritmo *Branch-and-Prune*. Já o algoritmo SBP implementa a proposta de simetria apresentada.

Os códigos foram implementados nas linguagens C/C++ utilizando o compilador GCC 7.4.0 no sistema operacional Ubuntu 7.4.0-1ubuntu1 18.04.1 com Intel® Core™ i3-7100U, 2.40 GHz e 16 GB de RAM.

Todos os códigos foram executados até que uma solução fosse encontrada ou o tempo máximo de 500 segundos fosse atingido. As instâncias utilizadas foram todas artificiais e formadas unicamente por nitrogênios (N), carbonos (C) e carbonos alfa ( $C_\alpha$ ) da cadeia principal (*backbone*).

Os dados (as arestas) que preenchem a matriz de distância  $E$  atenderam aos seguintes critérios:

$$|i - j| < 4 \text{ e } d(i, j) < 5\text{\AA},$$

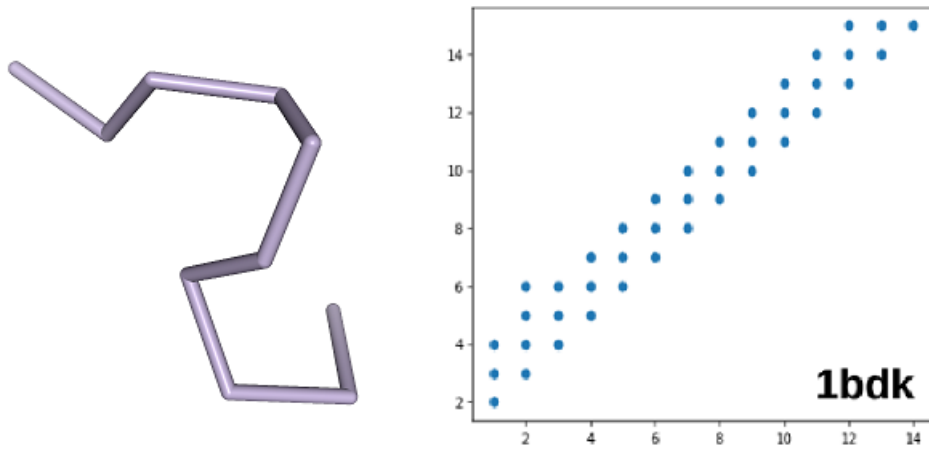
onde os átomos  $i$  e  $j$  são carbonos e nitrogênios. Ou seja, todo átomo de carbono e nitrogênio tem conhecidas as distâncias aos três átomos imediatamente consecutivos (arestas de discretização), assim como também todo par de átomos de carbono cuja distância seja inferior a  $5\text{\AA}$  (arestas de poda).

A seguir temos a visualização da matriz de conectividade e da estrutura tridimensional para três das instâncias utilizadas nos testes. Em seguida são apresentados os resultados, cujas medições são o tempo computacional e uma medida de erro chamada LDE (*Largest Distance Error*) definida como

$$LDE = \frac{1}{|E|} \sum_{(i,j) \in E} \frac{|||X_i - X_j|| - d_{ij}|}{d_{ij}}, \quad (6.1)$$

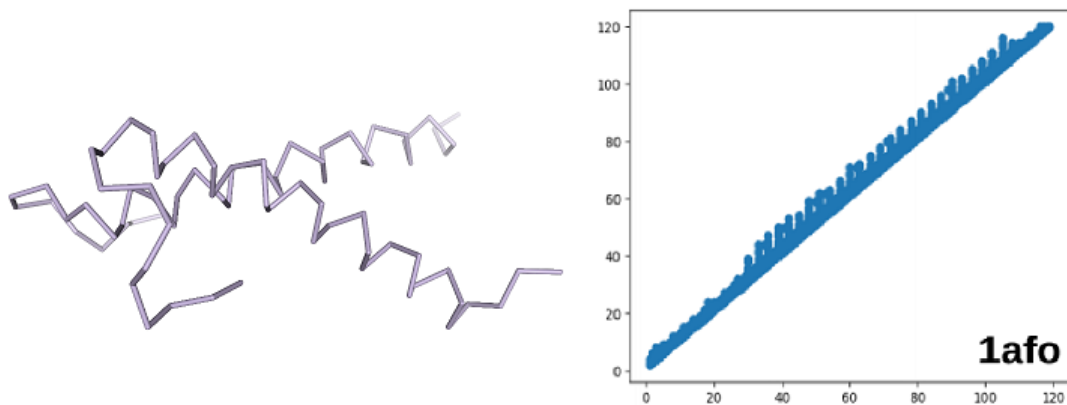
ou seja, é a diferença entre as posições obtidas pelo algoritmo e aquelas que são esperadas. Sendo esta uma medida de precisão das soluções (LIBERTI *et al.*, 2008).

Figura 27 – 1bdk: 15 vértices e 40 arestas.



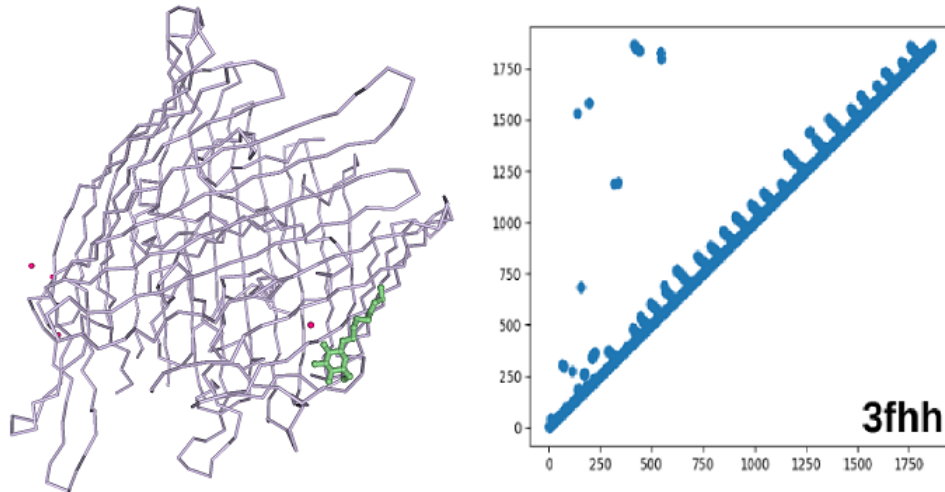
Fonte: Protein Data Bank (NATIONAL..., 2020)

Figura 28 – 1afo: 120 vértices e 518 arestas.



Fonte: Protein Data Bank (NATIONAL..., 2020)

Figura 29 – 3fhh: 1863 vértices e 7796 arestas.



Fonte: Protein Data Bank (NATIONAL..., 2020)

Tabela 1 – Tempo Computacional e LDE dos Experimentos Numéricos

Proteína	Vértices	Arestas	Tempo Computacional		LDE	
			MDJEEP	SBP	MDJEEP	SBP
1bfw	12	30	1.60E-05	6.92E-07	2.23E-16	6.09E-16
1bdk	15	40	2.50E-05	1.28E-06	2.53E-16	4.07E-16
1f3r	27	91	5.70E-05	6.41E-06	2.46E-16	8.06E-16
1bcv	57	187	9.20E-05	8.55E-06	3.75E-06	2.96E-13
1afo	120	518	1.99E-04	2.42E-05	1.82E-06	1.97E-12
1b4r	240	1016	4.97E+00	9.52E-04	5.79E-14	3.39E-12
1a3s	474	2043	1.84E+01	1.74E-03	5.25E-07	2.13E-11
1all	480	2199	1.06E-01	1.74E-04	1.98E-07	2.45E-12
1bob	918	3858	1.48E+02	1.46E-02	1.48E-13	2.38E-11
3fhh	1863	7796	–	2.66E-01	–	6.50E-11

## 7 CONCLUSÕES E TRABALHOS FUTUROS

A Geometria de Distâncias é uma área de estudos que envolve uma grande variedade de conceitos da Matemática e da Ciência da Computação, além de várias aplicações com características próprias. Estas aplicações, juntamente com as propriedades do problema, como por exemplo a simetria das estruturas, fornecem subsídios para novas abordagens de resolução do DGP.

Neste trabalho, aplicamos o conceito de simetria do  $KDMDGP$  de maneira original na fase de construção da primeira solução e não apenas na etapa pós-processamento como sugerido em (LAVOR *et al.*, ). O algoritmo SBP proposto a partir desta abordagem obteve melhor performance, quando comparado ao algoritmo BP, devido à redução da quantidade de operações realizadas. A melhora na performance do SBP, quando comparado ao BP, foi até 1000 vezes melhor mantendo a mesma qualidade das soluções com erro LDE equiparável.

O DGP é uma área de pesquisa e aprendizagem riquíssima e repleta de problemas envolventes e desafiadores. Dentre os projetos futuros, podemos tentar incorporar informações das matrizes de distância de soluções conhecidas e disponíveis em bancos de dados como o PDB para construir uma distribuição de probabilidade para os elementos faltantes na matriz de distância do DGP. Deste modo, a busca na árvore de possibilidades seria orientada por conhecimentos prévios e não incluídos na instância em particular.



## REFERÊNCIAS

- BAJAJ, C. The algebraic degree of geometric optimization problems. **Discrete & Computational Geometry**, [S.I.]: Springer, v. 3, n. 2, p. 177–191, 1988. Disponível em: <<https://doi.org/10.1007/BF02187906>>. Acesso em: 15 dez. 2020.
- BERMAN, H. M.; WESTBROOK, J.; FENG, Z. *et al.* The protein data bank. **Nucleic acids research**, Oxford University Press, v. 28, n. 1, p. 235–242, 2000.
- BISWAS, P. **Semidefinite programming approaches to distance geometry problems**. 2007. 125 f. Dissertation (Doctorate in Philosophy) – Stanford University, California, 2007. v. 68. Disponível em: <[http://www.optimization-online.org/DB\\_FILE/2008/12/2170.pdf](http://www.optimization-online.org/DB_FILE/2008/12/2170.pdf)>. Acesso em: 14 dez. 2020.
- BISWAS, P.; TOH, K.-C.; YE, Y. A distributed sdp approach for large-scale noisy anchor-free graph realization with applications to molecular conformation. **SIAM Journal on Scientific Computing**, [S.I.], v. 30, n. 3, p. 1251–1277, 2008.
- BLUMENTHAL, L. M. **Theory and applications of distance geometry**. [S.I.]: Oxford: Clarendon, 1953. 347p.
- CAMARGO, V. S. d. **Álgebra geométrica conforme e geometria de distâncias**. 2015. 129 f. Tese (Doutorado) – Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, Campinas, 2015. Disponível em: <<http://repositorio.unicamp.br/jspui/handle/REPOSIP/306798>>. Acesso em: 14 dez. 2020.
- CRIPPEN, G. M.; HAVEL, T. F. *et al.* **Distance geometry and molecular conformation**. [S.I.]: Taunton, Somerset, England: Research Studies Press; New York: Wiley, 1988.
- DEVLIN, T. M. **Manual de bioquímica com correlações clínicas**. [S.I.]: São Paulo: Editora Edgard Blücher, 1998.
- DONG, Q.; WU, Z. A linear-time algorithm for solving the molecular distance geometry problem with exact inter-atomic distances. **Journal of Global Optimization**, Springer, v. 22, n. 1-4, p. 365–375, 2002. Disponível em: <<https://doi.org/10.1023/A:1013857218127>>. Acesso em: 15 dez. 2020.
- DONG, Q.; WU, Z. A geometric build-up algorithm for solving the molecular distance geometry problem with sparse distance data. **Journal of Global Optimization**, Springer, v. 26, n. 3, p. 321–333, 2003. Disponível em: <<https://doi.org/10.1023/A:1023221624213>>. Acesso em: 15 dez. 2020.
- EREN, T.; GOLDENBERG, O.; WHITELEY, W.; YANG, Y. R. *et al.* Rigidity, computation, and randomization in network localization. In: IEEE INFOCOM 2004. HONG KONG. **Anais [...]**. [S.I.], IEEEExplore, 2004. v. 4, p. 2673–2684.
- GONCALVES, D. S.; LAVOR, C.; LIBERTI, L. *et al.* A new algorithm for the  $k$ dmdgp subclass of distance geometry problems. **arXiv.org**, [S.I.]: Cornell University, v.1, 11 set. 2020. Disponível em: <<https://arxiv.org/abs/2009.05404v1>>. Acesso em: 15 dez. 2020.
- LAVOR, C.; LIBERTI, L. Um convite à geometria de distâncias. **SBMAC: Notas em Matemática Aplicada**, São Carlos: SBMAC, v. 71, p. 56, 2014.
- LAVOR, C.; LIBERTI, L.; DONALD, B. *et al.* Minimal nmr distance information for rigidity of protein graphs. **Discrete Applied Mathematics**, Elsevier, v. 256, p. 91–104, 2019. Disponível em: <<https://doi.org/10.1016/j.dam.2018.03.071>>. Acesso em: 15 dez. 2020.
- LAVOR, C.; LIBERTI, L.; LODWICK, W. A. *et al.* **An Introduction to Distance Geometry applied to Molecular Geometry**. [S.I.]: Springer, 2017.

LAVOR, C.; LIBERTI, L.; MACULAN, N. *et al.* The discretizable molecular distance geometry problem. **Computational Optimization and Applications**, Springer, v. 52, n. 1, p. 115–146, 2012. Disponível em: <<https://doi.org/10.1007/s10589-011-9402-6>>. Acesso em: 15 dez. 2020.

LAVOR, C. C.; MACULAN, N.; SOUZA, M. F. D. *et al.* **Álgebra e geometria no cálculo de estruturas moleculares**. [S.l.]: Rio de Janeiro: IMPA, 2017. (31° Colóquio Brasileiro de Matemática).

LIBERTI, L.; LAVOR, C.; MACULAN, N. A branch-and-prune algorithm for the molecular distance geometry problem. **International Transactions in Operational Research**, Wiley Online Library, v. 15, n. 1, p. 1–17, 2008. Disponível em: <<https://doi.org/10.1111/j.1475-3995.2007.00622.x>>. Acesso em: 15 dez. 2020.

LIBERTI, L.; LAVOR, C.; MACULAN, N. *et al.* Euclidean distance geometry and applications. **Siam Review**, SIAM, v. 56, n. 1, p. 3–69, 2014. Disponível em: <<https://doi.org/10.1137/120875909>>. Acesso em: 15 dez. 2020.

LIBERTI, L.; LAVOR, C.; MUCHERINO, A. *et al.* Molecular distance geometry methods: from continuous to discrete. **International Transactions in Operational Research**, Wiley Online Library, v. 18, n. 1, p. 33–51, 2011. Disponível em: <<https://doi.org/10.1111/j.1475-3995.2009.00757.x>>. Acesso em: 15 dez. 2020.

LINDEGREN, L.; LAMMERS, U.; HOBBS, D. *et al.* The astrometric core solution for the gaia mission-overview of models, algorithms, and software implementation. **Astronomy & Astrophysics: Astronomical instrumentation**, EDP Sciences, v. 538, n. A78, 2012. Disponível em: <<https://doi.org/10.1051/0004-6361/201117905>>. Acesso em: 15 dez. 2020.

MENGER, K. Untersuchungen über allgemeine metrik. **Mathematische Annalen**, Springer, v. 103, n. 1, p. 466–501, 1930. Disponível em: <<https://doi.org/10.1007/BF01455705>>. Acesso em: 15 dez. 2020.

MORÉ, J. J.; WU, Z. Distance geometry optimization for protein structures. **Journal of Global Optimization**, Springer, v. 15, n. 3, p. 219–234, 1999. Disponível em: <<https://doi.org/10.1023/A:1008380219900>>. Acesso em: 15 dez. 2020.

NATIONAL SCIENCE FOUNDATION. **RCSB PDB (Protein Data Bank)**. U.S, 2020. Disponível em: <<https://www.rcsb.org>>. Acesso em: 15 dez. 2020.

NEUMAIER, A. Molecular modeling of proteins and mathematical prediction of protein structure. **SIAM review**, v. 39, n. 3, p. 407–460, 1997. Disponível em: <<https://doi.org/10.1137/S0036144594278060>>. Acesso em: 15 dez. 2020.

NIELSEN, J.; ROTH, B. On the kinematic analysis of robotic mechanisms. **The International Journal of Robotics Research**, SAGE Journals, v. 18, n. 12, p. 1147–1160, 1999. Disponível em: <<https://doi.org/10.1177/02783649922067771>>. Acesso em: 15 dez. 2020.

PHILLIPS, A. T.; ROSEN, J. B.; WALKE, V. H. Molecular structure determination by convex, global underestimation of local energy minima. **DIMACS Series in Discrete Mathematics and Theoretical Computer Science**, Citeseer<sup>x</sup>, v. 23, p. 181–198, 1994. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/versions?doi=10.1.1.81.1199>>. Acesso em: 15 dez. 2020.

SAXE, J. B. Two papers on graph embedding problems. **Journal Contributions**, Pittsburgh, Pennsylvania: Carnegie-Mellon University, 1980. Disponível em: <<https://doi.org/10.1184/R1/6611156.v1>>. Acesso em: 15 dez. 2020.

SCHLICK, T. Molecular modeling and simulation: an interdisciplinary guide: an interdisciplinary guide. **Interdisciplinary Applied Mathematics**, New York: Springer-Verlag, v. 21, n. 2, 2010.

SOUZA, M.; XAVIER, A. E.; LAVOR, C. *et al.* Hyperbolic smoothing and penalty techniques applied to molecular structure determination. **Operations Research Letters**, Elsevier, v. 39, n. 6, p. 461–465, 2011. Disponível em: <<https://doi.org/10.1016/j.orl.2011.07.007>>. Acesso em: 15 dez. 2020.

SOUZA, M. F. Suavização hiperbólica aplicada à otimização de geometria molecular. 2010. 86 f. Tese (Doutorado em Engenharia de Sistemas e Computação) – Universidade Federal do Rio de Janeiro, 2010. Disponível em: <<https://www.cos.ufrj.br/uploadfile/1265727560.pdf>>. Acesso em: 15 dez. 2020.