



**Universidade do Estado do Rio de Janeiro**  
**Instituto Politécnico**

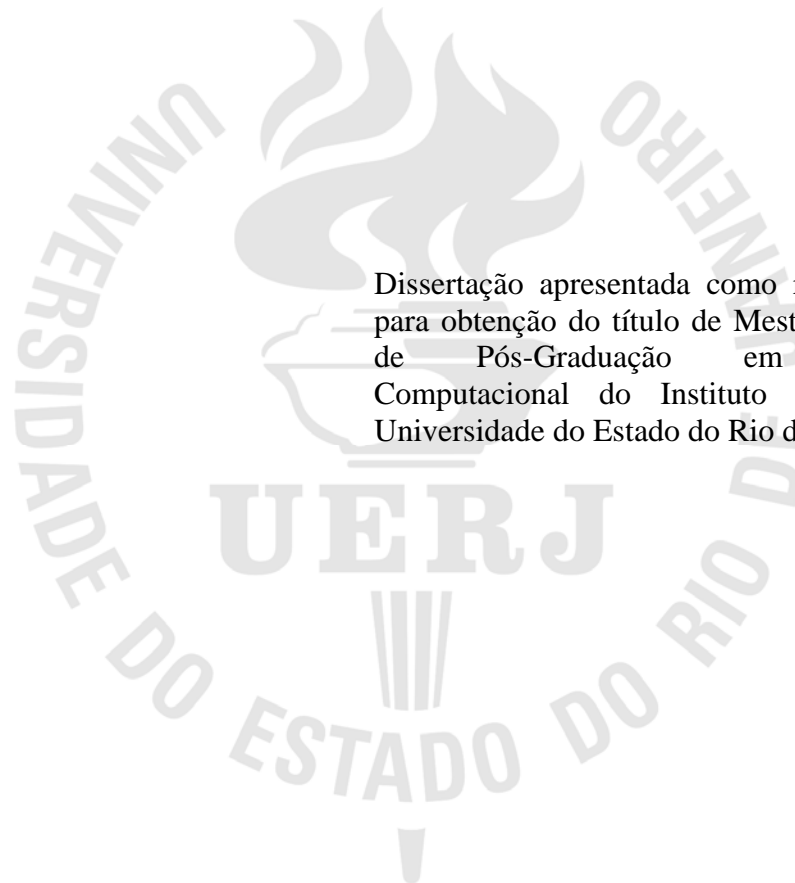
**Jardel da Silva Costa**

**Minimização do Potencial  
de Lennard-Jones Via  
Otimização Global**

**Nova Friburgo  
2010**

**Jardel da Silva Costa**

**Minimização do Potencial de  
Lennard-Jones Via Otimização Global**



Dissertação apresentada como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Modelagem Computacional do Instituto Politécnico, da Universidade do Estado do Rio de Janeiro.

Orientadores: Prof. Luiz Nélio Henderson Guedes de Oliveira, D. Sc.  
Prof. Wagner Figueiredo Sacco, D. Sc.

**Nova Friburgo  
2010**

CATALOGAÇÃO NA FONTE  
UERJ / REDE SIRIUS / BIBLIOTECA CTC/E

C837

Costa, Jardel da Silva.

Minimização do potencial de Lennard-Jones via  
otimização global / Jardel da Silva Costa. - 2010.  
53 f. : il.

Orientadores: Luiz Nélio Henderson Guedes de  
Oliveira e Wagner Figueiredo Sacco.

Dissertação (Mestrado) - Universidade do Estado  
do Rio de Janeiro, Instituto Politécnico.

1. Otimização matemática – Teses. 2. Processo  
estocástico – Teses. 3. Algoritmos – Métodos de  
simulação – Teses. 4. Lennard – Jones, Potencial de –  
Teses. I. Oliveira, Luiz Nélio Henderson Guedes de.  
II. Sacco, Wagner Figueiredo. III. Universidade do  
Estado do Rio de Janeiro. Instituto Politécnico. IV. Título.

CDU 519.863

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta  
dissertação.

---

Assinatura

---

Data

Jardel da Silva Costa

**Minimização do Potencial de Lennard-Jones  
Via Otimização Global**

Dissertação apresentada como requisito para obtenção do título de Mestre, ao Programa de Pós - Graduação em Modelagem Computacional do Instituto Politécnico, da Universidade do Estado do Rio de Janeiro.

Aprovada em 20 de agosto de 2010.

Banca examinadora:

---

Prof. Luiz Nélio Henderson Guedes de Oliveira, D. Sc. (Orientador)  
Instituto Politécnico da Universidade do Estado do Rio de Janeiro

---

Prof. Roberto Pinheiro Domingos, D. Sc.  
Instituto Politécnico da Universidade do Estado do Rio de Janeiro

---

Prof. Anderson Alvarenga de Moura Meneses, D. Sc.  
Observatório Nacional

Nova Friburgo  
2010

## DEDICATÓRIA

*Aos meus pais.*

## **AGRADECIMENTOS**

Agradeço a todos que de alguma forma contribuíram para este trabalho.

Em especial aos professores Nélio Henderson e Wagner Sacco.

Aos colegas Juan Diego e Nelson.

Ao meus amigos Ronie Stutz e Adriana Bernardes.

Aos membros da banca examinadora.

A todos os funcionários e professores do IPRJ.

À minha avó Malvina.

À CAPES.

*"Eu quase nada sei. Mas desconfio de muita coisa."*

*(Guimarães Rosa).*

## RESUMO

COSTA, Jardel da Silva. Minimização do potencial de Lennard-Jones via otimização global. 2010. 53 f. Dissertação (Mestrado em Modelagem Computacional) – Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, 2010.

Devido à sua importância, o chamado problema de Lennard-Jones tem atraído pesquisadores de diversos campos da ciência pura e aplicada. Tal problema resume-se em achar as coordenadas de um sistema no espaço Euclidiano tridimensional, as quais correspondem a um mínimo de um potencial de energia. Esse problema desempenha um papel de fundamental importância na determinação da estabilidade de moléculas em arranjos altamente ramificados, como das proteínas. A principal dificuldade para resolver o problema de Lennard-Jones decorre do fato de que a função objetivo é não-convexa e altamente não-linear com várias variáveis, apresentando, dessa forma, um grande número de mínimos locais. Neste trabalho, foram utilizados alguns métodos de otimização global estocástica, onde procurou-se comparar os resultados numéricos dos algoritmos, com o objetivo de verificar quais se adaptam melhor à minimização do referido potencial. No presente estudo, abordou-se somente micro agrupamentos possuindo de 3 a 10 átomos. Os resultados obtidos foram comparados também com o melhores resultados conhecidos atualmente na literatura. Os algoritmos de otimização utilizados foram todos implementados em linguagem C++.

**Palavras-chave:** Problema de Lennard-Jones; Otimização global; Métodos estocásticos.



## ABSTRACT

Because of its importance, the so-called Lennard-Jones problem has attracted researchers from various fields of pure and applied science. This problem boils down to find the coordinates of a system with three-dimensional Euclidean space, which correspond to minimum potential energy. This problem plays a fundamental role in determining the stability of molecules in highly branched arrangement, such as proteins. The main difficulty in solving the problem of Lennard-Jones from the fact that the objective function is non-convex and highly nonlinear with several variables, thus presenting a large number of local minima. Here, we used some methods of stochastic global optimization, where we seek to compare the results of the numerical algorithm, in order to see which are better suited to the minimization of the potential. In this study, we addressed only micro groups having 3-10 atoms. The results were also compared with the currently best known results in literature. The optimization algorithms were all implemented in C++.

**Keywords:** Lennard-Jones problem; Global optimization; Stochastic optimization methods.

## LISTA DE FIGURAS

Figura 2.1 - Diferentes deformações de uma molécula.....	19
Figura 3.1 - Potencial LJ de um único par de átomos.....	24
Figura 5.1: Luus-Jaakola pseudocódigo.....	31
Figura 5.2: PCA pseudocódigo.....	34
Figura 7.1: Função de Easom.....	38
Figura 7.2: Função de Himmelblau .....	39
Figura 7.3: Função Alpine com $D=2$ .....	39
Figura 8.1: Micro aglomerados estáveis (4 a 10 átomos). .....	47
Figura 8.2: Micro aglomerados estáveis (12 a 20 átomos). .....	48

## LISTA DE TABELAS

Tabela 7.1 — Parâmetros utilizados para os testes.....	40
Tabela 7.2 — Resultados obtidos com o algoritmo Luus-Jaakola para os testes.....	40
Tabela 7.3 — Resultados obtidos com o algoritmo SA para os testes.....	41
Tabela 7.4 — Resultados obtidos com o algoritmo PCA para os testes.....	41
Tabela 7.5 — Resultados obtidos com o algoritmo ED para os testes.....	42
Tabela 8.1 — Resultados obtidos com o algoritmo Luus-Jaakola (3 a 6 átomos).....	43
Tabela 8.2 — Resultados obtidos com o algoritmo Luus-Jaakola (7 a 10 átomos).....	43
Tabela 8.3 — Resultados obtidos com o algoritmo SA (3 a 6 átomos).....	44
Tabela 8.4 — Resultados obtidos com o algoritmo SA (7 a 10 átomos).....	44
Tabela 8.5 — Resultados obtidos com o algoritmo PCA (3 a 6 átomos).....	45
Tabela 8.6 — Resultados obtidos com o algoritmo PCA (7 a 10 átomos).....	45
Tabela 8.7 — Resultados obtidos com o algoritmo ED (3 a 6 átomos).....	46
Tabela 8.8 — Resultados obtidos com o algoritmo ED (7 a 10 átomos).....	46

## LISTA DE ABREVIATURAS E SIGLAS

ED	Evolução Diferencial
FO	Função Objetivo
LJ	Lennard-Jones [LJ também é como usualmente se abrevia o Luus-Jaakola]
PCA	Particle Collision Algorithm
SA	Simulated Annealing

## LISTA DE SÍMBOLOS

### *Símbolos para o problema de Lennard-Jones*

$N$	Número de átomos
$V_{MM}$	Energia potencial.
$V_b$	Energia de ligação
$V_\theta$	Ângulo de flexão de energia
$V_\tau$	Energia de torção;
$V_{nb}$	Energia de interação entre dois átomos não-ligados
$k_{r,i}$	Força de ligação constante;
$r_i$	Comprimento de ligação;
$r_i^o$	Comprimento de ligação na posição de equilíbrio
$n$	Número total de ligações na molécula
$D$	Caracterização do vínculo
$a$	Caracterização do vínculo
$k_{\theta,ij}$	Força constante
$\theta_{ij}$	Ângulo de ligação entre os 3 átomos e
$\theta_{ij}^o$	Ângulo de ligação na posição de equilíbrio entre os 3 átomos.
$A_{i,n}$	Força constante que controla a amplitude da curva;
$\Phi$	Fase;
$n$	Periodicidade de $A_{i,n}$
$\tau_i$	Ângulo de torção
$r_{ij}$	Distância entre dois átomos de não-ligados $i$ e $j$
$A_{ij}$	Constante de Van der Waals
$B_{ij}$	Constante de Van der Waals
$y^i$	Coordenadas do $i$ -ésimo átomo
$y^j$	Coordenadas do $j$ -ésimo átomo

### *Símbolos dos algoritmos de otimização*

$\vec{X}$	Vetor
$X^*$	Melhor ponto do espaço de busca
$\max f(\vec{X})$	Valor máximo assumido pela função $f$
$\min f(\vec{X})$	Valor mínimo assumido pela função $f$
$\varepsilon$	Coefficiente de contração do Luus-Jaakola
$n_{in}$	Número de loops internos
$n_{out}$	Número de loops externos
$\delta f$	Aumento de $f$
$T$	Temperatura do <i>Simulated Annealing</i>
$p$	Probabilidade do <i>Simulated Annealing</i>
$np$	Número de indivíduos da população
$X_i$	Indivíduo $i$ da população
$Y_i$	Indivíduo $i$ da população de mutantes
$Z_i$	Indivíduo $i$ da população pós cruzamento
$F$	é uma taxa que pondera a mutação
$CR$	probabilidade de cruzamento
$f(\ )$	representa a avaliação do indivíduo

### *Outros símbolos*

$R^3$	Conjunto do Espaço Euclidiano
$>$	Maior que
$<$	Menor que
$\sum$	Somatório
$\ x\ $	Norma do vetor $x$

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>17</b>
<b>2</b>	<b>MECÂNICA MOLECULAR.....</b>	<b>19</b>
<b>3</b>	<b>O POTENCIAL DE LENNARD-JONES.....</b>	<b>24</b>
<b>3.1</b>	<b>Formulação.....</b>	<b>25</b>
<b>4</b>	<b>ABORDAGEM DO PROBLEMA.....</b>	<b>27</b>
<b>5</b>	<b>ALGORITMOS DE OTIMIZAÇÃO.....</b>	<b>29</b>
<b>5.1</b>	<b>Algoritmo de Luus-Jaakola.....</b>	<b>30</b>
<b>5.2</b>	<b>Simulated Annealing.....</b>	<b>32</b>
<b>5.3</b>	<b>Particle Collision Algorithm, PCA.....</b>	<b>33</b>
<b>5.4</b>	<b>Evolução Diferencial.....</b>	<b>35</b>
<b>6</b>	<b>IMPLEMENTAÇÃO.....</b>	<b>37</b>
<b>6.1</b>	<b>Ambiente de Testes Para as Implementações.....</b>	<b>37</b>
<b>6.2</b>	<b>Critérios de Avaliação dos Métodos de Otimização.....</b>	<b>37</b>
<b>6.3</b>	<b>Metodologia Experimental.....</b>	<b>37</b>
<b>7</b>	<b>RESULTADOS PRELIMINARES COM FUNÇÕES-TESTE.....</b>	<b>38</b>
<b>7.1</b>	<b>Funções-teste.....</b>	<b>38</b>
<b>7.1.1</b>	<b>Exemplo 1 – Easom.....</b>	<b>38</b>
<b>7.1.2</b>	<b>Exemplo 2 – Himmelblau.....</b>	<b>38</b>
<b>7.1.3</b>	<b>Exemplo 3 – Alpine.....</b>	<b>39</b>
<b>7.2</b>	<b>Resultados preliminares.....</b>	<b>40</b>
<b>7.2.1</b>	<b>Algoritmo Luus-Jaakola.....</b>	<b>40</b>
<b>7.2.2</b>	<b>Algoritmo <i>Simulated Annealing</i>.....</b>	<b>41</b>
<b>7.2.3</b>	<b>Algoritmo PCA.....</b>	<b>41</b>
<b>7.2.4</b>	<b>Algoritmo Evolução Diferencial.....</b>	<b>42</b>

<b>8</b>	<b>RESULTADOS</b> .....	43
<b>8.1</b>	<b>Algoritmo Luus-Jaakola</b> .....	43
<b>8.2</b>	<b>Algoritmo <i>Simulated Annealing</i></b> .....	44
<b>8.3</b>	<b>Algoritmo PCA</b> .....	45
<b>8.4</b>	<b>Algoritmo Evolução Diferencial</b> .....	46
<b>9</b>	<b>CONCLUSÕES E PERSPECTIVAS</b> .....	49
<b>9.1</b>	<b>Conclusões</b> .....	49
<b>9.2</b>	<b>Perspectivas</b> .....	50
	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> .....	51



## 1 INTRODUÇÃO

O problema de Lennard-Jones é tido como extremamente simples de descrever, no entanto é muito difícil resolvê-lo (WILLE; VENNIK, 1985). Em química computacional, um problema de conformação molecular pode ser descrito como a busca pelo mínimo global de um potencial de energia apropriado que depende das posições relativas dos átomos. Boas técnicas para a minimização desse potencial facilitam a concepção de drogas, síntese e utilização de medicamentos e de produtos (KUBINYI et al., 1998). É bem conhecido que o sucesso da modelagem computacional deste tipo de problema depende de dois fatores: (1) uma função potencial adequada para prever os estados de mínima energia do sistema e (2) algoritmos de otimização capazes de localizar o mínimo global do potencial de energia utilizado na modelagem.

Moléculas consistem de elétrons e núcleos. A maioria dos processos da química quântica separa o movimento dos núcleos do movimento dos elétrons. Este tratamento é chamado de aproximação de Born-Oppenheimer. Essa aproximação resulta em um modelo de núcleos movendo-se em uma superfície de energia, com elétrons ajustando-se instantaneamente às mudanças de posições nucleares. O movimento nuclear é limitado pela interação de núcleos e elétrons. Em quaisquer posições fixas dos núcleos, o potencial de energia é a soma da repulsão entre os núcleos de carga positiva e atrações decorrentes dos elétrons. Tal energia pode ser calculada resolvendo-se a equação de Schrödinger, da mecânica quântica. O equilíbrio geométrico de uma molécula (comprimentos de ligação e ângulos) está associado às coordenadas de um mínimo da superfície do potencial de energia.

A metodologia da química quântica pode ser usada para prever a geometria e a energia de moléculas. No entanto, em termos de tempo de computador, tal metodologia é, em geral, muito cara, mesmo para casos simples, podendo se tornar intratável no caso de muitas moléculas orgânicas ou estruturas biológicas macromoleculares (DOUCET; WEBER, 1996). Portanto, na prática, se torna importante a utilização de modelos capazes de fornecer uma conformação energética favorável para grandes sistemas, em um tempo razoável. Nesse sentido, a chamada mecânica molecular disponibiliza técnicas que desempenham um papel importante na investigação da conformação de muitas moléculas de interesse prático.

Neste trabalho, nosso objetivo central é analisar o desempenho de alguns algoritmos baseados em metaheurísticas na minimização global do potencial de Lennard-Jones, um modelo básico e bem conhecido da mecânica molecular. Tais algoritmos são utilizados em suas formas canônicas, a fim de proporcionar a escolha de métodos com estruturas básicas mais promissoras para a modelagem computacional.

Dessa forma, o esforço despendido na presente dissertação pode ser visto como parte do trabalho inicial para a concepção de algoritmos híbridos que utilizam heurísticas seguidas de buscas determinísticas, em regiões mais apropriadas para a obtenção de minimizadores globais.

## 2 MECÂNICA MOLECULAR

A mecânica molecular concebe as moléculas como um sistema massa-mola, onde as massas são os átomos e as molas representam as ligações moleculares. Na Figura 2.1, mostramos a visualização das diferentes deformações (principalmente a torção, o vínculo de alongamento e do ângulo de flexão) operando em uma molécula simples.

A deformação devido à interação entre dois átomos não ligados representa a ação de atração de Van der Waals, a repulsão estérica, resultante de uma força entrópica repulsiva pela sobreposição das moléculas poliméricas, e atração-repulsão eletrostática sobre estes dois átomos.

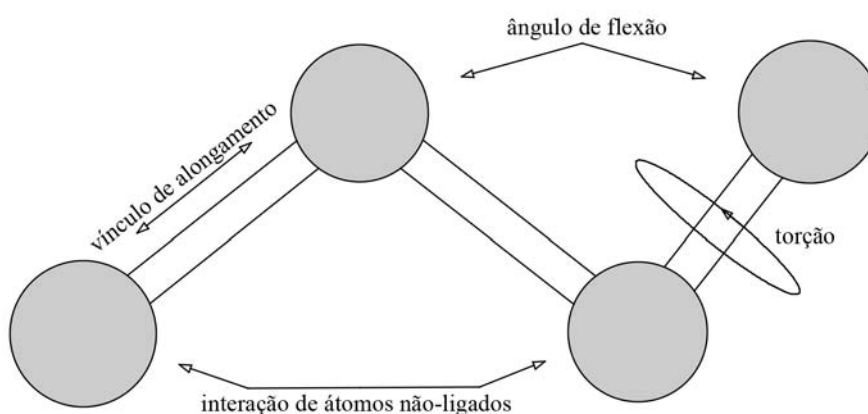


Figura 2.1: diferentes deformações de uma molécula.

O objetivo da modelagem molecular é prever a energia associada a uma dada conformação de uma molécula. A energia de uma molécula-alvo depende das posições relativas de seus átomos. Esta energia pode ser aproximadamente estimada pela soma de diversas contribuições:

$$V_{MM} = V_b + V_\theta + V_\tau + V_{nb} + (\text{termos específicos}) \quad (2.1)$$

onde  $V_{MM}$  é muitas vezes referida como a energia potencial. Ela corresponde à diferença de energia entre a molécula real e uma molécula hipotética em que todos os valores estruturais, tais como comprimentos de ligação e ângulos de ligação são exatamente ideais (valores de equilíbrio).

Na equação (2.1),  $V_b$  representa a energia de ligação, descrevendo a compressão ou a extensão de uma ligação de seu comprimento de equilíbrio;

$V_\theta$  representa o ângulo de flexão de energia;

$V_\tau$  é a energia de torção;

$V_{nb}$  é a energia de interação entre dois átomos não-ligados.

A contribuição do vínculo de alongamento é representada pela Lei de Hooke. Ela mede a energia devido à variação do comprimento de ligação após a extensão ou compressão de seus valores de equilíbrio:

$$V_b = \frac{1}{2} \sum_{i=1}^n k_{r,i} (r_i - r_i^o)^2 \quad (2.2)$$

onde

$k_{r,i}$  é a força de ligação constante;

$r_i$  é o comprimento de ligação;

$r_i^o$  é o comprimento de ligação na posição de equilíbrio e

$n$  é o número total de ligações na molécula.

Os parâmetros  $k_{r,i}$  e  $r^o$  são invariáveis, dependendo apenas do tipo de cada par de átomos ligados. Equação (2.2) é uma aproximação grosseira da energia de ligação. Como alternativa, um potencial de Morse pode ser usado para descrever mais justamente o vínculo de alongamento de energia devido à variação do comprimento de uma ligação:

$$V_b = \sum_{i=1}^n D \left( 1 - e^{-a(r_i - r_i^o)} \right)^2 \quad (2.3)$$

onde os parâmetros  $D$  e  $a$  são características do vínculo. O uso de tal potencial parece ser útil para ligações de hidrogênio de forma alongada (ALLINGER, 1985).

O ângulo de flexão determina a quantidade do potencial energético que resulta pela variação do ângulo entre duas ligações adjacentes com base em um ângulo de ligação de equilíbrio. No caso de aproximação harmônica, este é igualmente derivado da lei de Hooke:

$$V_\theta = \frac{1}{2} \sum_{ij} k_{\theta,ij} (\theta_{ij} - \theta_{ij}^o)^2 \quad (2.4)$$

onde

$k_{\theta,ij}$  é a força constante;

$\theta_{ij}$  é o ângulo de ligação entre os 3 átomos e

$\theta_{ij}^o$  é o ângulo de ligação na posição de equilíbrio entre os 3 átomos.

A energia de torção representa a energia de modificação da rotação da molécula em torno de um vínculo. A expressão mais comum que permite descrever a avaliação da energia molécula como a função de ângulo de torção é a série de Fourier:

$$V_\tau = \frac{1}{2} \sum_{i=1}^n A_{i,n} (1 + \cos(n\tau_i - \Phi)) \quad (2.5)$$

onde

$A_{i,n}$  é a força constante que controla a amplitude da curva;

$\Phi$  é a fase;

$n$  é a periodicidade de  $A_{i,n}$  e

$\tau_i$  é o ângulo de torção.

A energia de torção é na verdade uma correção de diferentes termos de energia, em vez de um processo físico. Ela representa a quantidade de energia que deve ser adicionado ou subtraído do somatório  $V_b + V_\theta + V_{nb}$ , a fim de se obter a geometria de acordo com um experimento ou com a geometria que é deduzida dos cálculos da química quântica.

Já a energia de interação não-ligação provém das interações entre dois átomos não ligados. Sua principal causa é o impedimento estérico, que desempenha um importante papel na geometria molecular.

A energia de interação é a soma das energias de todos os átomos não ligados entre si. Ela inclui a energia de interação de Van der Waals, energia eletrostática e três termos de energia de indução.

O termo de interação de Van der Waals é geralmente descrito pelo potencial de Lennard-Jones (LJ) (LENNARD-JONES, 1924):

$$V_{vdw} = \sum_{i < j} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right) \quad (2.6)$$

onde

$r_{ij}$  é a distância entre dois átomos não-ligados  $i$  e  $j$ ,  $A_{ij}$  e  $B_{ij}$  são constantes de Van der Waals.

A soma é tomada sobre todos os pares não-ligados de átomos  $(i, j)$ .

Essas expressões envolvem dois termos:

1. Uma parte atrativa, correspondente ao dipolo indução ou interação, proporcional a  $r_{ij}^{-6}$ , onde  $r_{ij}$  é a distância entre os dois átomos  $i$  e  $j$ .
2. Uma parte repulsiva, que corresponde aos termos da dispersão de London (EISENSCHITZ; LONDON, 1930), que cresce rapidamente quando a distância diminui.

Para um dado arranjo geométrico dos átomos em um sistema molecular, a energia estérica, devida às distorções de comprimentos de ligação e ângulos com respeito aos valores de referência e interação de Van der Waals, pode ser calculada de acordo com a função de

potencial de energia. Para determinar o equilíbrio geométrico real, esta energia estérica com respeito a todos os graus internos de liberdade deve ser minimizada.

O termo de energia eletrostática aumenta com a polaridade dos compostos químicos. Ele pode ser expresso pelo potencial de Coulomb. A energia de indução é a consequência da distorção da distribuição eletrônica, que depende do campo elétrico criado por outras moléculas.

Comprimentos de ligação e ângulos de ligação são habitualmente disponíveis nas informações estruturais, isto é, da cristalografia de raios X. Parâmetros de alongamento podem ser diretamente obtidos de constantes de força vibracional. Os coeficientes das barreiras de torção podem ser estimados a partir de alturas de barreira obtidos através de espectroscopia de microondas, de estudos da termodinâmica, ou raios infravermelhos e espectroscopia Raman. Mais difícil é a avaliação da interação de Van der Waals, um ponto crucial, já que essas interações são importantes na determinação da estabilidade do aglomerado de moléculas altamente ramificadas como peptídeos.

### 3 O POTENCIAL DE LENNARD-JONES

Conforme descrito no capítulo anterior, o potencial de Van der Waals caracteriza a contribuição dos pares não-ligados entre os átomos. Ele é geralmente descrito pela função potencial LJ, uma vez introduzida pela Equação (2.5). O potencial LJ é uma parte fundamental de muitos modelos de energia, incluindo todas as funções de energia comumente utilizada para as proteínas. Um sistema que contenha mais de um átomo, cuja interação de Van der Waals pode ser descrita pelo potencial LJ é chamado de agrupamento LJ. Em nosso trabalho, utilizamos o seguinte potencial:

$$V_{vdw} = \sum_{i < j} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right) \quad (3.1)$$

A função potencial LJ de um único par de átomos neutros é uma simples função unimodal. Isto é ilustrado pela Figura 3.1

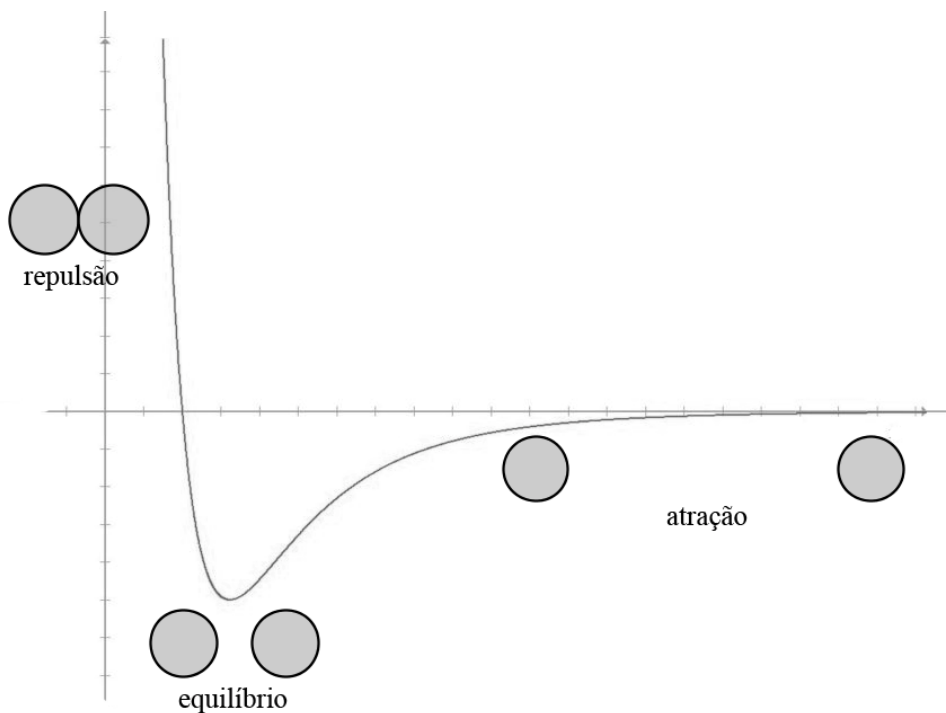


Figura 3.1: potencial LJ de um único par de átomos.



É fácil encontrar o mínimo global desta função que assume energia -1. Em um sistema complexo, muitos átomos interagem e precisamos somar as funções do potencial LJ para cada par de átomos em um agrupamento. O resultado é uma estrutura energética complexa, com muitos mínimos locais. Dado um agrupamento, por exemplo, de três átomos, cada átomo interage com dois outros átomos. O potencial LJ pode ser escrito como:

$$V = \left( \frac{1}{r_{12}^{12}} - \frac{2}{r_{12}^6} \right) + \left( \frac{1}{r_{13}^{12}} - \frac{2}{r_{13}^6} \right) + \left( \frac{1}{r_{23}^{12}} - \frac{2}{r_{23}^6} \right) = \sum_{1 \leq i < j \leq 3} \left( \frac{1}{r_{ij}^{12}} - \frac{2}{r_{ij}^6} \right) \quad (3.2)$$

Note que usamos no somatório  $i < j$ , porque precisamos contar a interação entre dois átomos de uma única vez. Se usarmos  $i \neq j$ , o somatório de energia deve ser dividido por 2. Em outras palavras, a função potencial LJ é parcialmente separável, ou seja, é uma função que é a soma de outras, cada uma das quais envolve apenas um subconjunto disjunto das variáveis.

A separabilidade parcial do potencial LJ implica que, se um único átomo ou molécula em um grupo é movido, a energia potencial pode ser re-avaliada em um custo mais barato que é de apenas  $\frac{2}{N}$  -ésimo do custo de uma avaliação total da função, onde  $N$  é o número total de átomos ou moléculas no conjunto. Isto é devido ao fato de que a função é composta pela soma de pares de interações entre átomos ou moléculas.

Dado um conjunto de  $N$  átomos, o problema do agrupamento de Lennard-Jones consiste em achar a posição dos átomos no espaço euclidiano de três dimensões que representa um mínimo de energia potencial.

### 3.1 Formulação

Considere que  $y^i = (y_1^i, y_2^i, y_3^i)^T$  representa as coordenadas do átomo  $i$  no espaço Euclidiano tridimensional. Seja  $Y = ((y_1)^T, \dots, (y_N)^T)^T$ , onde  $N$  é o número de átomos em agrupamento.

O potencial de LJ de um par de átomos  $(i, j)$  é

$$v(r_{ij}) = \frac{1}{r_{ij}^{12}} - \frac{2}{r_{ij}^6}$$

$$\text{onde } r_{ij} = \|y^i - y^j\|$$

O problema LJ descrito na seção anterior pode ser formulado no espaço de coordenadas da seguinte forma:

$$\begin{aligned} \min_{Y \in R^{3N}} V(Y) &= \sum_{i < j} v(\|y^i - y^j\|) \\ &= \sum_{i=1}^{N-1} \sum_{j=i+1}^N \left( \frac{1}{\|y^i - y^j\|^{12}} - \frac{2}{\|y^i - y^j\|^6} \right) \end{aligned}$$

onde  $y^i$  e  $y^j$  representam as coordenadas do  $i$ -ésimo e do  $j$ -ésimo átomo, respectivamente.

Como foi ilustrado pela Figura 1.2, para um único par de átomos neutros, o mínimo global de energia potencial é alcançado quando a distância entre dois átomos é 1. Quando esta distância se aproxima de zero, o potencial tenderá ao infinito. Longe do sistema, a contribuição do átomo para potencial total torna-se quase nula. Devido a estas observações, é razoável esperar que na solução ótima do problema de LJ todos os átomos em  $R^3$  estão perto de uma unidade de distância uns dos outros. Foi demonstrado que a complexidade de determinar o mínimo global de energia do cluster LJ pertence à classe de problema NP-difícil, (WILLE; VENNIK, 1985). Em outras palavras, não existe algoritmo conhecido que possa resolver esse problema em tempo polinomial.

## 4 ABORDAGEM DO PROBLEMA

Várias abordagens computacionais para o problema do potencial de Lennard-Jones podem ser encontrados na literatura, variando de modelos de programação matemática (GOCKENBACH et al., 1997), algoritmos genéticos, (DEAVEN et al., 1996) e método de Monte Carlo, (WALES; DOYE, 1997). Bons resultados foram obtidos pelo acoplamento de alguns destes métodos. Um exemplo é o chamado método de dinâmica de Langevin, (BISWAS; HAMANN, 1986). Este método é uma combinação de *Simulated Annealing* e de técnicas de gradiente e tem provado ser suficiente para minimizar a energia total dos sistemas de pequeno porte.

A principal dificuldade em resolver o problema decorre do fato de que a função objetivo é não-convexa com muitas variáveis e possui um grande número de mínimos locais. Esta não-convexidade dificulta a determinação da solução global.

Um algoritmo apropriado para a solução deste tipo de problema deve atentar para alguns aspectos geométricos, a fim de lidar com os vales estreitos apresentados pela função objetivo. Isso, recentemente, tem motivado a utilização de métodos de busca direta para a otimização global do potencial de Lennard-Jones.

Nesta dissertação, empregamos alguns algoritmos baseados em metaheurísticas para a otimização global do potencial de Lennard-Jones. O desempenho de tais algoritmos é comparado visando, principalmente, hibridizações futuras com algoritmos de busca local para a minimização de problemas com muitas variáveis.

Dentre muitas possibilidades de escolha dos métodos de otimização, optamos por métodos heurísticos de naturezas distintas. Assim sendo, foram selecionados o algoritmo Luus-Jaakola (LUUS; JAAKOLA, 1973), por sua peculiaridade de contrair o espaço de busca em torno das melhores soluções, o algoritmo *Simulated Annealing* (METROPOLIS et al., 1958) e o algoritmo PCA (SACCO; OLIVEIRA, 2005), por serem do tipo Metropolis, ou seja, por permitirem, com uma certa probabilidade, que um resultado pior que o melhor corrente seja aceito, e finalmente o algoritmo Evolução Diferencial (STORN; PRICE, 1995), por ser do tipo populacional.

Os próximos capítulos estão organizados da seguinte forma: no Capítulo 5, apresentamos os algoritmos de otimização, cujos detalhes de implementação são discutidos no

Capítulo 6. O Capítulo 7 mostra resultados preliminares com algumas funções testes. Os resultados da aplicação desses algoritmos ao problema de Lennard-Jones são descritos no Capítulo 8. O Capítulo 9 contém as conclusões e também sugestões para pesquisas futuras.

## 5 ALGORITMOS DE OTIMIZAÇÃO

Em Otimização podemos utilizar algoritmos que busquem encontrar uma configuração de variáveis ou informações que sejam a melhor solução de um problema. A cada configuração analisada é designado um “prêmio”, mais conhecido como “*fitness*”, que quantifica a adequação daquela configuração àquele problema, sendo a *fitness* nos problemas mais simples o valor de uma função real, chamada função objetivo. Espera-se que a solução do problema seja a configuração com a melhor *fitness* encontrada.

Uma solução possível é frequentemente representada por um vetor  $\vec{X} = (x_0, x_1, \dots, x_n)$ , onde os termos  $x_i$  são suas componentes, e o número  $n$  é o número de variáveis ou dimensões do problema em questão. A aptidão de cada solução viável é medida por uma função objetivo ou função custo dada por  $f = f(\vec{X})$  onde a solução ótima global do sistema é referenciada por  $X^*$  de modo que  $X^* = \max f(\vec{X})$  ou  $X^* = \min f(\vec{X})$ .

Os algoritmos clássicos de otimização, particularmente os determinísticos, frequentemente necessitam dos valores das derivadas da função objetivo para efetuarem a busca pelo ótimo, ou seja, a função deve ser diferenciável. Há uma classe de algoritmos que não usam as derivadas da função objetivo. Tais algoritmos são chamados de métodos de busca direta, os quais são mais versáteis, posto que podem atacar problemas que não são expressos por uma função diferenciável, ou mesmo resolver problemas que não podem ser descritos por uma função matemática com formulação clássica, como comumente encontrados no contexto da engenharia e das ciências aplicadas. Dentre esses existem os métodos estocásticos em que a busca pelo ótimo é guiada por processos aleatórios. Abordagens de otimização global podem ser classificadas como probabilísticas ou determinísticas. Métodos como *Simulated Annealing* e algoritmos genéticos são métodos probabilísticos, já os métodos do tipo Gradiente, *quasi-Newton* e Branch and Bound (LAND; DOIG, 1960) são determinísticos.

Segundo o “*No Free lunch theorem*” enunciado por (WOLPERT; MACREADY, 1997), dado um algoritmo  $a$  executado  $m$  vezes em um problema  $p$ , o desempenho desse algoritmo é definido como uma probabilidade para se encontrar um resultado satisfatório. Assim o somatório de tais probabilidades para todos os problemas de otimização é sempre igual para todos os algoritmos. Portanto justifica-se a aplicação de diferentes algoritmos a um

mesmo problema, pois um algoritmo altamente especializado em uma classe de problemas poderá apresentar desempenho inferior em outra classe.

Os métodos utilizados para encontrar o mínimo global do potencial de LJ são na maioria das vezes baseados em heurísticas, que incluem a mimetização dos fenômenos físicos, buscas aleatórias combinadas com abordagens de otimização local. Neste capítulo, apresentaremos algumas técnicas de otimização estocástica. Os algoritmos que são utilizados na nossa abordagem são descritos em detalhes.

### 5.1 Algoritmo de Luus-Jaakola

Métodos de busca para otimização que se baseiam na exploração aleatória de um domínio ou espaço de busca a fim de encontrar-se um ponto ou solução que minimize uma função objetivo foram introduzidos por (ANDERSON, 1953), depois desenvolvidos por (KARNNOP, 1963) e (MATYAS, 1965).

Vários algoritmos nesta vertente foram propostos para otimização contínua, principalmente no campo da engenharia química: o método de Luus e Jaakola (1973), o método de Gaines e Gaddy (GAINES; GADDY, 1976) e o método de Salcedo (SALCEDO et. al., 1990), entre outros. Dentre estes métodos, o mais conhecido é o Luus-Jaakola o qual tem sido aplicado a vários campos como engenharia química (LUUS; JAAKOLA, 1973, LEE et al., 1999, LUUS; HENNESSY, 1999), problemas de controle (LUUS, 2001), de óptica (AL-MARZOUG; HODGSON, 2006) engenharia elétrica (SINGH, 2005), de cromatografia (POPLEWSKA et al., 2006) e engenharia nuclear (SACCO et al., 2007) além de outras aplicações.

Sua idéia central consiste em considerar uma região ampla que englobe os possíveis valores das variáveis, valores factíveis, e geram-se soluções aleatórias enquanto a região de busca torna-se menor ao longo das iterações. A Figura 5.1 a seguir exhibe um pseudocódigo.

```

Escolha um espaço de busca inicial  $r^{(0)}$ , um número de loops externos  $n_{out}$  e
um número de loops internos  $n_{in}$ .
Escolha um coeficiente de contração  $\varepsilon$ .
Gere aleatoriamente uma solução inicial  $X^*$ 
Para  $i = 1$  to  $n_{out}$ 
    Para  $j = 1$  to  $n_{in}$ 
         $X^{(j)} = X^* + R^{(j)}r^{(i-1)}$ , onde  $R^{(j)}$  é uma matriz diagonal de números
aleatórios entre -0.5 e 0.5.
        Se  $\text{Fitness}(X^{(j)}) < \text{Fitness}(X^*)$ 
             $X^* = X^{(j)}$ 
        fim Se
    fim para
     $r^{(i)} = (1 - \varepsilon)r^{(i-1)}$ 
fim para

```

Figura 5.1: Luus-Jaakola pseudocódigo

Em síntese podemos dividir o algoritmo em dois passos:

1. Dado um número especificado de iterações calcule  $x = x^* + Ar$ , onde  $x^*$  é o ótimo corrente,  $A$  é uma matriz diagonal com elementos aleatórios (distribuição uniforme) entre -0,5 e +0,5 e  $r$  é um vetor contendo os raios das regiões de busca para as variáveis do problema (definido pelo usuário);
2. Ao final do número de iterações especificadas no passo 1,  $x^*$  é substituído pelo melhor valor obtido no passo anterior, e a região de busca é contraída através de uma expressão do tipo  $r(j+i) = (1 - \varepsilon)r(j)$ , onde  $\varepsilon$  é um número próximo de zero.

Notemos que o algoritmo de Luus-Jaakola, ao contrário da maioria dos métodos de otimização estocásticos, não aceita valores que provoquem aumento de função-objetivo. Sua capacidade de localização do ótimo global reside na propriedade de dar "grandes saltos", ou seja, produzir um novo ótimo distante do ótimo corrente. Esta capacidade é controlada pelo tamanho da região de busca, contraída após um número fixo de iterações. Contrariamente à essência do método de Luus-Jaakola, o algoritmo *Simulated Annealing*, por exemplo,

funciona produzindo novos pontos que sejam vizinhos ao ótimo local, em uma estrutura markoviana, como veremos a seguir.

## 5.2 *Simulated Annealing*

Como o próprio nome indica, o Arrefecimento Simulado é baseado no processo de resfriamento lento de um metal até o seu estado de solidificação, o qual ocorre em um mínimo global da sua energia, onde evidencia-se uma estrutura cristalina. Em geral, tal processo é simulado para a busca de um mínimo global de uma função objetivo. O algoritmo SA foi inicialmente proposto por (METROPOLIS et al., 1958) para achar o equilíbrio da configuração de um conjunto de átomos em uma dada temperatura. A ligação entre este algoritmo e a minimização matemática foi inicialmente observada por (PINCUS, 1970), mas foi (KIRKPATRICK et al., 1983) que a usou como base de uma técnica de otimização combinatória.

A grande vantagem do *Simulated Annealing* sobre outros métodos é a sua capacidade de evitar que a solução seja aprisionada em um mínimo local. O algoritmo utiliza uma busca aleatória, que não só aceita as alterações que diminuem a função objetivo, mas também algumas alterações que a aumentam com uma probabilidade  $p = e^{\frac{-\delta f}{T}}$  onde  $\delta f$  é o incremento de  $f$  e  $T$  é um parâmetro de controle, que por analogia com a aplicação original é conhecido como a "temperatura".

Descreveremos brevemente como o SA funciona. Dada uma função a ser minimizada e alguns valores iniciais para as variáveis, inicia-se uma avaliação em um nível elevado de temperatura. Durante o resfriamento a temperatura cai de acordo com uma taxa de resfriamento estipulada. Para que o SA funcione satisfatoriamente, o resfriamento tem que ser lento, repetidamente escolhe-se um subconjunto das variáveis que são modificados aleatoriamente em uma determinada vizinhança do ponto atual. Se a função objetivo possui um valor de função mais baixo na nova iteração, os novos valores são escolhidos para serem os valores iniciais para a próxima iteração. Se o valor da função é maior, os novos valores são escolhidos para serem os valores iniciais para a próxima iteração com certa probabilidade, dependendo da variação do valor da função objetivo e da temperatura. É este esquema de seleção estocástica, conhecido como Paradigma de Metropolis, que garante a não-convergência prematura para mínimos locais.



A mudança para um estado com um valor de função objetivo pior dá-se com maior probabilidade a temperaturas mais altas, quando o algoritmo vasculha uma fatia maior do espaço de busca. Durante todo este processo, a temperatura é diminuída gradativamente, até que os valores não mudem mais. Então, a função atingiu o seu mínimo, podendo ser o global.

Com inspiração do *Simulated Annealing* têm sido desenvolvidos novos algoritmos que também fazem uso do paradigma de Metropolis como o PCA que veremos a seguir.

## 5.2 Particle Collision Algorithm, PCA

O algoritmo de colisão de partículas (Particle Collision Algorithm) foi desenvolvido por SACCO e DE OLIVEIRA (2005) é um algoritmo do tipo Metropolis em que o único parâmetro necessário é o número de avaliações da função objetivo. Este algoritmo foi inspirado na física das reações de colisão de partículas nucleares, particularmente absorção e espalhamento.

No PCA, uma partícula que atinge uma região com altos valores de função objetivo (no caso de maximização), é “absorvida” explorando os arredores desta região. Por outro lado, uma partícula que atinge uma região com baixos valores de função objetivo é espalhada para outras regiões do espaço de busca. Com certa probabilidade, pode haver ainda exploração de uma região ruim.

Os passos do PCA seguem na figura 5.2:

<p>Gerar uma solução aleatória <b><i>old</i></b> entre um limite inferior e um limite superior do domínio de busca considerado. A função <i>rand()</i> é um gerador de número aleatório em um intervalo.</p> <p><b>Passo principal</b></p> <p>1- Gerar uma perturbação estocástica <b><i>new</i></b> de <b><i>old</i></b>;</p> <p>2- Se <math>f(\mathbf{new}) &gt; f(\mathbf{old})</math> ou <math>f(\mathbf{new}) = f(\mathbf{old})</math> então ir para a Exploração; Senão ir para o Espalhamento;</p> <p>Fim.</p>
<p><b>Exploração</b> (fazer por 100 vezes)</p> <p>1- Gerar uma pequena perturbação estocástica <b><i>new</i></b> de <b><i>old</i></b>:</p> <p>Upper <math>\leftarrow rand(1,0 ; 1,2) * \mathbf{old}</math>;</p> <p>lower <math>\leftarrow rand(0,8 ; 1,0) * \mathbf{old}</math>;</p> <p><math>\mathbf{new} \leftarrow \mathbf{old} + (\text{upper} - \mathbf{old}) * rand(0,0 ; 1,0) - (\mathbf{old} - \text{lower}) * (1 - rand(0,0 ; 1,0))</math>;</p> <p>checar os limites de domínio.</p> <p>2- Se <math>f(\mathbf{new}) &gt; f(\mathbf{old})</math> então <math>\mathbf{old} \leftarrow \mathbf{new}</math>;</p> <p>Fim da exploração.</p>
<p><b>Espalhamento</b></p> <p>1- Faça <math>p = 1 - f(\mathbf{new}) / f(\mathbf{old})</math>;</p> <p>2- Se <math>p &gt; rand(0,0 ; 1,0)</math> então <math>\mathbf{old} \leftarrow rand(\text{limite inferior} ; \text{limite superior})</math>;</p> <p>Senão vá para a exploração.</p> <p>Fim do espalhamento.</p>

Figura 5.2: PCA pseudocódigo

O PCA é um algoritmo robusto e de fácil implementação, pois só tem um parâmetro que é o número de iterações. Isto é uma boa vantagem em relação a outras metaheurísticas que comumente necessitam de vários parâmetros. Segundo a literatura, (SACCO; OLIVEIRA, 2005),  $10^4$  a  $10^6$  iterações tem se mostrado um valor suficiente.

### 5.3 Evolução Diferencial

A evolução diferencial (STORN; PRICE, 1995), é um algoritmo bioinspirado desenvolvido para otimização. Neste algoritmo cada solução factível é vista como elemento de uma população com  $np$  indivíduos, a função é avaliada em todos os indivíduos e através de uma heurística cria-se uma nova população, repete-se a avaliação e faz-se uma comparação com a população anterior. Isto ocorre por um número determinado de vezes. A seguir descreveremos suas etapas.

Geramos uma população inicial aleatória de soluções factíveis cobrindo uniformemente o espaço de busca e executamos três operadores:

**Mutação:** criamos novos indivíduos, denotados por vetores modificados, pela adição da diferença ponderada entre dois indivíduos aleatórios da população a um terceiro indivíduo. Este operador pode ser executado da seguinte forma:

$$Y_i = X_{r_1} + F \cdot (X_{r_2} - X_{r_3}), \quad 1 \leq i \leq np$$

Onde  $Y_i$  é o vetor mutante gerado pelos vetores  $X_{r_1} \neq X_{r_2} \neq X_{r_3}$  e  $0 \leq F \leq 2$  é uma taxa que pondera a mutação. Esta operação é similar a recombinação aritmética dos algoritmos genéticos com representação real ou de ponto flutuante. Dessa forma são criados  $np$  mutantes.

**Cruzamento:** neste passo podemos preservar indivíduos sem que sofram a mutação, o objetivo é manter a diversidade da população e evitar convergências prematuras. Assim criamos de acordo com uma probabilidade  $CR$  uma população  $Z$  de  $np$  indivíduos que contém mutantes e não-mutantes da seguinte forma:

$$Z_i = \begin{cases} Y_i, & \text{rand}[0,1] \leq CR \\ X_i, & \text{senão} \end{cases}, \quad 1 \leq i \leq np$$

Onde  $\text{rand}[0,1]$  é um número aleatório no intervalo  $[0,1]$  e  $CR$  é um parâmetro que determina a probabilidade de um mutante ser aceito, assim  $CR = 1$  implica que todos os mutantes são aceitos e a cada iteração toda a população é modificada e  $CR = 0$  implica que nenhum mutante é aceito.

**Seleção:** A seleção é o processo de escolher os melhores indivíduos. Diferentemente de outros algoritmos evolutivos, a evolução diferencial não usa hierarquia (elitismo) nem seleção proporcional. É um procedimento muito simples:

$$X_i = \begin{cases} Z_i, & f(Z_i) < f(X_i) \\ X_i, & \text{senão} \end{cases}, \quad 1 \leq i \leq np$$

Onde  $f(\ )$  representa a avaliação do indivíduo, assim se o indivíduo  $Z_i$  produzir um menor valor de função objetivo então  $X_i$  passa a ser  $Z_i$ , caso contrário  $X_i$  é mantido.

Terminada a seleção temos uma nova população  $X$  com  $np$  indivíduos e podemos repetir os passos até um critério de parada.

## 6 IMPLEMENTAÇÃO E METODOLOGIA

### 6.1 Ambiente de Testes Para as Implementações

O ambiente utilizado para as implementações dos algoritmos Luus-Jaakola, *Simulated Annealing*, PCA e Evolução Diferencial para o problema do Potencia de Lennard-Jones foi um PC Core 2 Quad, com 4 Gb de RAM rodando no sistema operacional Windows Vista onde os códigos foram compilados em linguagem C++ com o software Codeblocks.

Como gerador de números aleatórios utilizamos o *Mersenne twister*, (MATSUMOTO; NISHIMURA, 1998), que gera números com uma distribuição quase uniforme no intervalo  $2^{19937} - 1$  para 32 bits.

### 6.2 Critérios de Avaliação dos Métodos de Otimização

Como critérios de avaliação utilizados nos reportamos aos mais aceitos e consagrados na literatura (HEDER; FUKUSHIMA, 2006; HIRSH et. al., 2007, como exemplos):

- o valor médio relacionado a todos os resultados obtidos em um determinado número de execuções do algoritmo com uma determinada configuração específica;
- o numero máximo ou mínimo de avaliações da função objetivo necessário para obter um bom resultado para a configuração escolhida;
- finalmente a taxa de acertos ou a razão entre o número de vezes em que o ótimo obtido foi encontrado e o número total de execuções. Consideramos que o algoritmo alcançou o ótimo com um erro de 0,001.

### 6.3 Metodologia Experimental

Para todos os algoritmos utilizados foram feitas baterias com 100 execuções experimentais e com sementes aleatórias distintas. Assim podemos observar o nível de robustez que cada método possui.

Estipulou-se como critério de parada um número máximo de avaliações da função objetivo em um milhão de modo a dar para cada algoritmo um bom tempo computacional afim de que diminua nossas dúvidas acerca das possibilidades de revelar o ótimo. Em todos os casos um experimento nunca demorou mais do que 20 segundos.

## 7 RESULTADOS PRELIMINARES COM FUNÇÕES-TESTE

### 7.1 Funções-teste

Escolhemos três funções teste por suas peculiaridades para avaliar previamente como os algoritmos considerados se comportariam.

#### 7.1.1 Exemplo 1 - Easom

Easom é uma função unimodal, onde o mínimo global está em uma área pequena em relação ao espaço de busca.

$$f(x_1, x_2) = -\cos(x_1) \cdot \cos(x_2) \cdot e^{-((x_1 - \pi)^2 + (x_2 - \pi)^2)}$$

$$-100 \leq x_i \leq 100, i = 1: 2$$

Ótimo -1 em  $(\pi, \pi)$

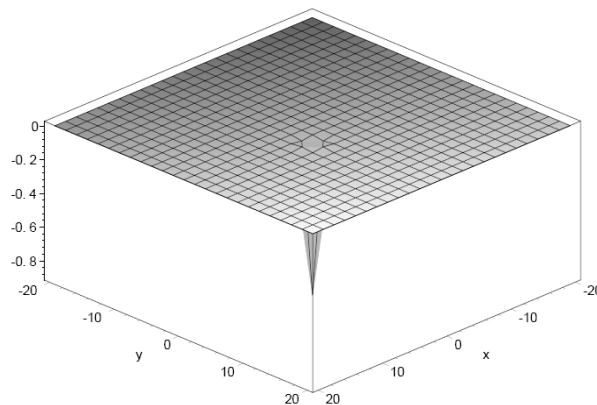


Figura 7.1: Função de Easom

#### 7.1.2 Exemplo 2 - Himmelblau

Este exemplo expresso pelas equações a seguir é um sistema retirado de (MARANAS; FLOUDAS, 1995) que possui 9 raízes, seu domínio é  $[-5, 5]^2$  e para todas as raízes e o ótimo é 0.

$$\begin{cases} f_1 = 4x_1^3 + 4x_1x_2 + 2x_2^2 - 42x_1 - 14 \\ f_2 = 4x_2^3 + 4x_1x_2 + 2x_1^2 - 26x_1 - 22 \end{cases}$$

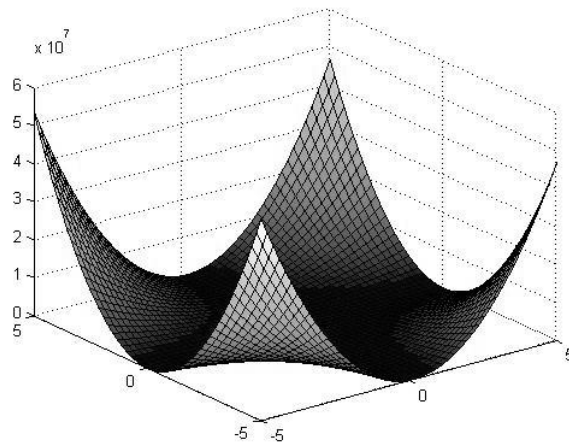


Figura 7.2: Função de Himmelblau

### 7.1.3 Exemplo 3 - Alpine

A função Alpine (CLERC, 1999) é uma função multimodal dada por

$$f(x) = \prod_{i=1}^n \sin(x_i) \sqrt{\prod_{i=1}^n x_i}, x_i \in [0, 10]^D$$

para alguma dimensão  $D$ , seu máximo global é aproximadamente  $2.808^D$  no ponto que é solução de  $\tan(x) + 2x = 0, x \in \left] \frac{5\pi}{2}, \frac{6\pi}{2} \right[$ , ou seja o ponto  $(7.917, \dots, 7.917)$ .

Para duas dimensões temos a Figura 7.3. Esta função é interessante para testar a procura de um extremo, pelas seguintes razões:

- há muitos extremos locais;
- há apenas um extremo global;
- a solução pode ser facilmente calculada diretamente.

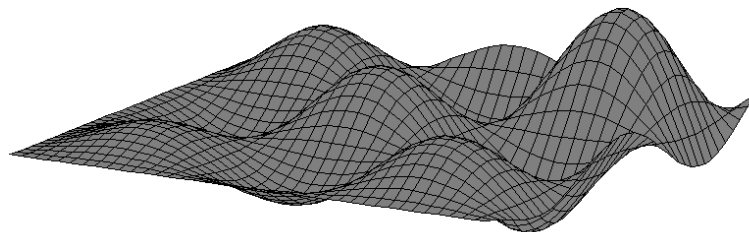


Figura 7.3: Função Alpine com  $D=2$

## 7.2 Resultados preliminares

Utilizamos os parâmetros da tabela 7.1 a seguir para os testes com os algoritmos.

Tabela 7.1 — Parâmetros utilizados para os testes

Luus-Jaakola	PCA	<i>Simulated Annealing</i>	Evolução Diferencial
$n_{out} = 50000$ $n_{in} = 20$ $\varepsilon = 0.0001 \%$	N de avaliações = $10^6$	$T = 1000$ $\varepsilon = 0.999$ N de avaliações = $10^6$	Numero de indivíduos = 1000 $F = 0.5$ $CR = 0.8$ Número de avaliações da FO = $10^6$

### 7.2.1 Algoritmo Luus-Jaakola

Tabela 7.1 — Resultados obtidos com o algoritmo Luus-Jaakola para os testes

Função	Easom	Himmelblau	Alpine	Alpine
Número de dimensões do espaço de busca	2	2	2	8
Ótimo Global	-1	0	7.884864	3865.243255
Melhor Valor da F.O.	-1	5.83372e-08	7.8856	-3866.68
Pior Valor da F.O.	-1	4.48936e-05	7.8856	857.416
Valor Médio da F.O.	-1	3.7042e-06	7.8856	2147.34
Desvio Padrão	1.16852e-08	5.70207e-06	7.28286e-09	842.356
Taxa de Acertos	100	100	100	15



7.2.2 Algoritmo *Simulated Annealing*

Tabela 7.2 — Resultados obtidos com o algoritmo SA para os testes

Função	Easom	Himmelblau	Alpine	Alpine
Número de dimensões do espaço de busca	2	2	2	8
Ótimo Global	-1	0	7,884864	3865,243255
Melhor Valor da F.O.	-0.999997	4.44479e-06	7.8856	2555.87
Pior Valor da F.O.	-0.999042	0.0333611	7.88503	1181.71
Valor Médio da F.O.	-0.999828	0.00663251	7.88546	1612.92
Desvio Padrão	0.000185868	0.00640499	0.000129714	279.212
Taxa de Acertos	41	16	100	0

## 7.2.3 Algoritmo PCA

Tabela 7.3 — Resultados obtidos com o algoritmo PCA para os testes

Função	Easom	Himmelblau	Alpine	Alpine
Número de dimensões do espaço de busca	2	2	2	8
Ótimo Global	-1	0	7,884864	3865,243255
Melhor Valor da F.O.	-1	-1.981e-09	7.8856	3247.65
Pior Valor da F.O.	-0.998	-0.0142832	7.88556	920.263
Valor Médio da F.O.	-0.999705	-0.00015063	7.88559	1646.09
Desvio Padrão	0.000350493	0.00142762	9.85232e-06	460.969
Taxa de Acertos	38	98	100	0

## 7.2.4 Algoritmo Evolução Diferencial

Tabela 7.4 — Resultados obtidos com o algoritmo ED para os testes

Função	Easom	Himmelblau	Alpine	Alpine
Número de dimensões do espaço de busca	2	2	2	12
Ótimo Global	-1	0	7,884864	240306
Melhor Valor da F.O.	-1	0.0571526	7.8856	240441
Pior Valor da F.O.	-1	34.2143	7.8856	240441
Valor Médio da F.O.	-1	6.79702	7.8856	240441
Desvio Padrão	0	6.74785	1.33898e-14	0.0588449
Taxa de Acertos	100	0	100	100

## 8 RESULTADOS OBTIDOS

Para cada algoritmo listaremos a seguir os resultados encontrados para a melhor configuração de parâmetros obtida já utilizada na tabela 7.1. Apresentamos nas tabelas os ótimos globais conhecidos (WALES; DOYE, 1997).

### 8.1 Algoritmo Luus-Jaakola

Tabela 8.1 — Resultados obtidos com o algoritmo Luus-Jaakola (3 a 6 átomos)

Número de átomos	3	4	5	6
Número de dimensões do espaço de busca	9	12	15	18
Ótimo Global	-3,000000	-6,000000	-9,103900	-12,712100
Melhor Valor da F.O.	-3,000000	-6,000000	-9.10384	-12.712
Pior Valor da F.O.	-3,000000	-3.00174	-6.00052	-5.03593
Valor Médio da F.O.	-3,000000	-5.91285	-8.19872	-10.7518
Desvio Padrão	7.43203e-08	0.445812	1.33638	1.94551
Taxa de Acertos	100	95	68	4

Tabela 8.2 — Resultados obtidos com o algoritmo Luus-Jaakola (7 a 10 átomos)

Número de átomos	7	8	9	10
Número de dimensões do espaço de busca	21	24	27	30
Ótimo Global	-16,5054	-19,8215	-24,11334	-28,4225
Melhor Valor da F.O.	-16.5052	-19.8211	-24.1108	-27.5201
Pior Valor da F.O.	-7.00662	-5.00844	-7.21426	-8.19483
Valor Médio da F.O.	-12.3577	-13.9775	-15.4462	-17.1966
Desvio Padrão	2.57467	3.41975	3.43052	4.2705
Taxa de Acertos	9	3	0	0

## 8.2 Algoritmo *Simulated Annealing*

Tabela 8.3 — Resultados obtidos com o algoritmo SA (3 a 6 átomos)

Número de átomos	3	4	5	6
Número de dimensões do espaço de busca	9	12	15	18
Ótimo Global	-3,000000	-6,000000	-9,103900	-12,712100
Melhor Valor da F.O.	-2.99975	-5.69102	-8.03204	-8.02354
Pior Valor da F.O.	-2.95704	-4.50124	-4.95026	-5.56844
Valor Médio da F.O.	-2.99204	-4.87979	-5.8755	-6.41005
Desvio Padrão	0.00617995	0.246379	0.485795	0.507657
Taxa de Acertos	0	0	0	0

Tabela 8.4 — Resultados obtidos com o algoritmo SA (7 a 10 átomos)

Número de átomos	7	8	9	10
Número de dimensões do espaço de busca	21	24	27	30
Ótimo Global	-16,505400	-19,821500	-24,1133400	-28,422500
Melhor Valor da F.O.	-8.83851	-9.66662	-9.86985	-9.91613
Pior Valor da F.O.	-6.03762	-6.39909	-6.67039	-7.23592
Valor Médio da F.O.	-6.88562	-7.47126	-7.73129	-8.14004
Desvio Padrão	0.466565	0.615015	0.573728	0.557318
Taxa de Acertos	0	0	0	0

### 8.3 Algoritmo PCA

Tabela 8.5 — Resultados obtidos com o algoritmo PCA (3 a 6 átomos)

Número de átomos	3	4	5	6
Número de dimensões do espaço de busca	9	12	15	18
Ótimo Global	-3,000000	-6,000000	-9,103900	-12,712100
Melhor Valor da F.O.	-3,000000	-5.99622	-9.07662	-12.3225
Pior Valor da F.O.	-2.99975	-5.06882	-5.06977	-6.10419
Valor Médio da F.O.	-2.99992	-5.46524	-7.61794	-9.4193
Desvio Padrão	4.89057e-05	0.385632	0.994652	1.4249
Taxa de Acertos	71	0	0	0

Tabela 8.6 — Resultados obtidos com o algoritmo PCA (7 a 10 átomos)

Número de átomos	7	8	9	10
Número de dimensões do espaço de busca	21	24	27	30
Ótimo Global	-16,505400	-19,821500	-24,1133400	-28,422500
Melhor Valor da F.O.	15.261	16.1604	18.5374	20.3335
Pior Valor da F.O.	7.11701	7.17566	7.95499	8.30958
Valor Médio da F.O.	10.9436	11.5465	12.9325	14.0078
Desvio Padrão	2.03967	2.39789	2.52329	2.82945
Taxa de Acertos	0	0	0	0

## 8.4 Algoritmo Evolução Diferencial

Tabela 8.7 — Resultados obtidos com o algoritmo ED (3 a 6 átomos)

Número de átomos	3	4	5	6
Número de dimensões do espaço de busca	9	12	15	18
Ótimo Global	-3,000000	-6,000000	-9,103900	-12,712100
Melhor Valor da F.O.	-3,000000	-6,000000	-9,103900	-12,712100
Pior Valor da F.O.	-3,000000	-6,000000	-9.10385	-11.2331
Valor Médio da F.O.	-3,000000	-6,000000	-9.08742	-9.73849
Desvio Padrão	0	0	0.00164335	0.319908
Taxa de Acertos	100	100	99	0

Tabela 8.8 — Resultados obtidos com o algoritmo ED (7 a 10 átomos)

Número de átomos	7	8	9	10
Número de dimensões do espaço de busca	21	24	27	30
Ótimo Global	-16,505400	-19,821500	-24,1133400	-28,422500
Melhor Valor da F.O.	-12.1675	-13.0039	-13.7609	-13.7297
Pior Valor da F.O.	-9.83336	-9.92327	-9.93733	-10.2425
Valor Médio da F.O.	-10.7128	-11.0305	-11.3701	-11.3436
Desvio Padrão	0.451139	0.556365	0.750673	0.699773
Taxa de Acertos	0	0	0	0

A seguir exibimos a título de ilustração a geometria esperada para que cada micro aglomerado tenha seu potencial de energia seja mínimo.

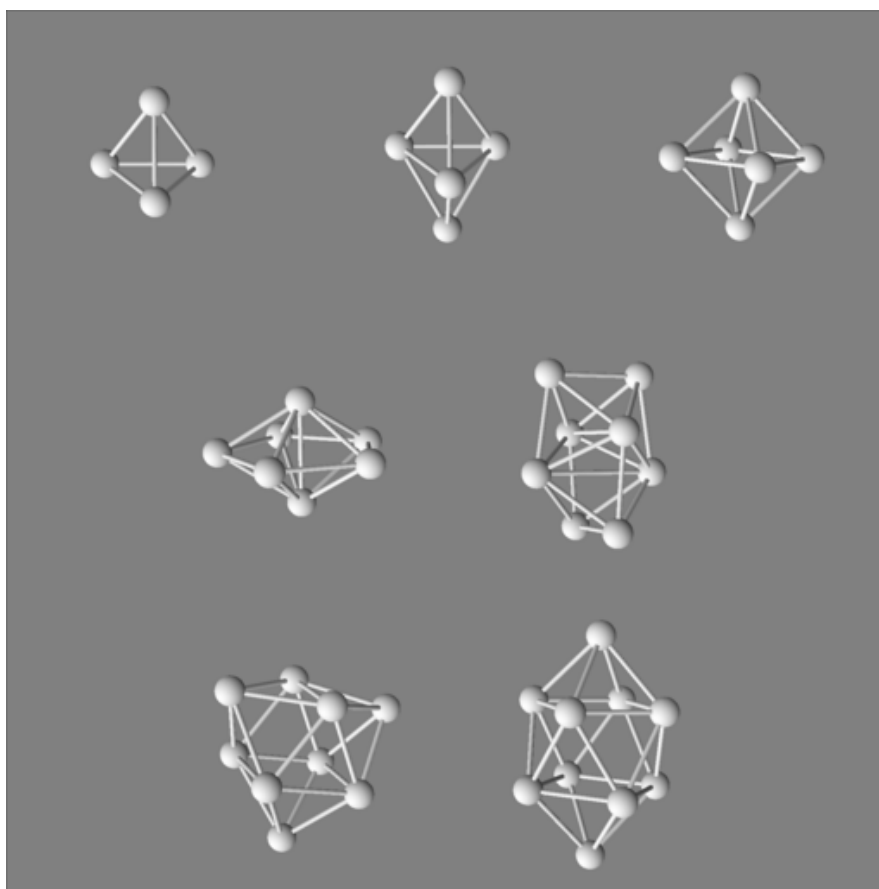


Figura 8.1: Micro aglomerados estáveis (4 a 10 átomos).

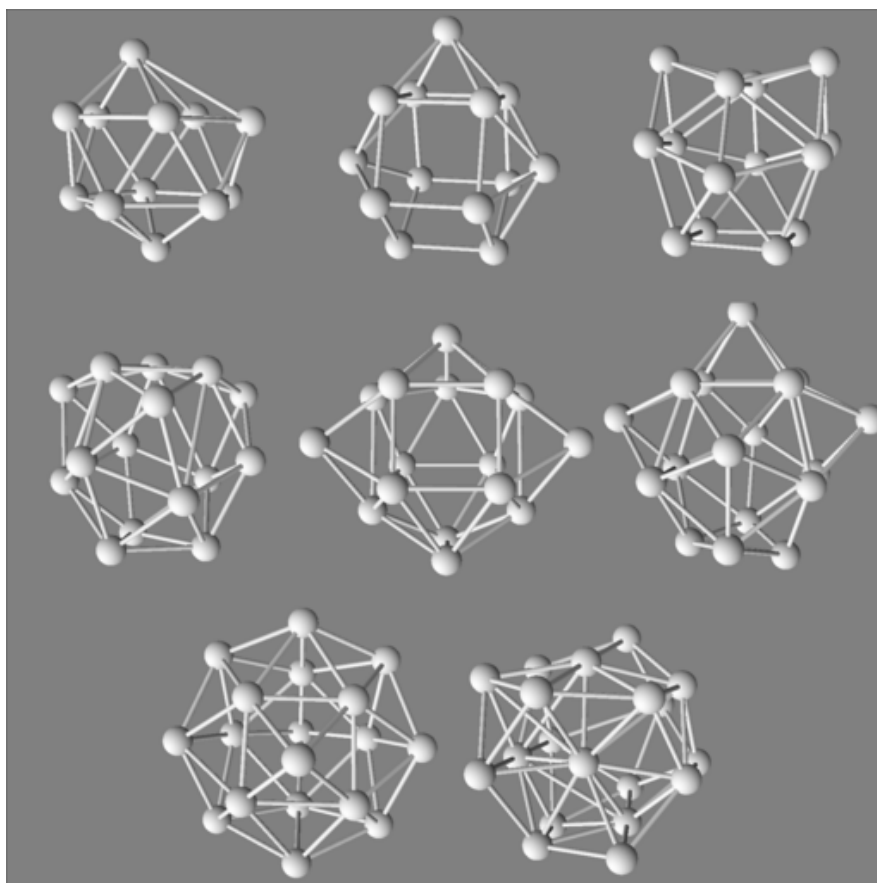


Figura 8.2: Micro aglomerados estáveis (12 a 20 átomos).



## 9 CONCLUSÕES E PERSPECTIVAS

### 9.1 Conclusões

Fizemos implementações com os algoritmos em suas formas canônicas sem a utilização de recursos sofisticados. Este esforço é um passo para a tomada de decisão acerca de quais métodos utilizar em hibridizações com algoritmos de busca local.

Como pudemos observar no Capítulo 7, um algoritmo pode resolver um problema de forma completamente satisfatória e ser incapaz de tratar outro, assim não podemos dizer que um algoritmo é melhor que outro e tornam-se necessários testes que verifiquem até que ponto o algoritmo é indicado para determinado problema.

De acordo com (JEZOWSKI; BOCHENEK, 2002) o Luus-Jaakola canônico pode se tornar deficiente para resolução de problemas multidimensionais, fato que pudemos verificar claramente na tabela 7.1.

Os algoritmos *Simulated Annealing* e PCA que aderem ao paradigma de Metropolis mostraram-se ineficientes para tratar a função de Easom que tem um ótimo global isolado em uma pequena região do domínio. Naturalmente o fato de permitirem que uma solução ruim seja aceita faz com que eles fujam da região do ótimo algumas vezes. Já para a função Himmelblau, que possui diversas regiões de ótimo, o PCA teve um desempenho melhor do que o *Simulated Annealing* como podemos ver nas tabelas 7.2 e 7.3. Em relação a esta mesma função Himmelblau notamos que o Evolução Diferencial tem um péssimo desempenho.

Todos os quatro algoritmos conseguiram encontrar o ótimo da função Alpine em duas dimensões, mas somente o Evolução Diferencial conseguiu tratar esta função de forma multidimensional.

Devemos salientar que para todas as funções teste utilizamos os mesmos parâmetros de configuração para os algoritmos. Sabemos que com algum ajuste desses parâmetros o algoritmo pode ter um desempenho melhorado ou não para o problema. Sendo assim optamos pelos mesmos parâmetros que foram utilizados para tratar o potencial de Lennard-Jones.

Em relação ao problema do potencial de Lennard-Jones notamos o seguinte:

- o Luus-Jaakola aproxima-se da região do ótimo para até 10 átomos;
- o *Simulated Annealing* aproxima-se para até 5 átomos;

- o PCA para até 7 átomos;
- o Evolução Diferencial para até 6 átomos.

Justamente o *Simulated Annealing* que é um algoritmo que vem sendo usado em hibridizações para resolução do problema de LJ (FAN; 2002) teve o pior desempenho. O Evolução Diferencial, apesar de tratar muito bem a multimodalidade da função teste Alpine, mostrou-se ineficaz para o problema LJ afastando-se fortemente da região do ótimo nos casos com mais de 7 átomos.

Para hibridizações com algoritmos de busca local concluímos que o Luus-Jaakola e o PCA seriam as melhores alternativas.

O Luus-Jaakola pelo fato de ter se aproximado sempre da região do ótimo. Notamos que essa aproximação reduz-se com o aumento do número de dimensões da função e por isso não podemos garantir que o Luus-Jaakola seja uma boa alternativa para um número muito grande de átomos. O Luus-Jaakola é muito simples e não tem qualquer heurística, foi o algoritmo mais rápido e teve um bom desempenho.

O PCA pelo fato de não ter se afastado tanto da região do ótimo como o Evolução diferencial, por exemplo, e ser semelhante, porém melhor, que o *Simulated Annealing* para tratar o problema.

## 9.2 Perspectivas

Já que parece muito interessante investir no Luus-Jaakola para uma hibridização com algoritmos de busca local.

Se hibridizações com *Simulated Annealing* conseguem bons resultados (FAN, 2002) esperamos que com o PCA alcançaremos um desempenho melhor ainda. A primeira proposta é utilizar o PCA hibridizado com técnicas de gradiente assim como (BISWAS; HAMANN, 1986) e a seguir testar com técnicas dos métodos de Newton, *quasi*-Newton e Nelder-Mead por exemplo.

**REFERÊNCIAS BIBLIOGRÁFICAS**

- AL-MARZOUG, S.M.; HODGSON, R.J.W. “Optimization of platinum-carbon multilayer mirrors for hard X-ray optics”, *Optics Communications*, 268, 84-89, 2006.
- ALLINGER, N.L., Molecular mechanics. In *International School of Crystallography: Static and Dynamic Implications of Precise Structural Information*, 149–164. Italy, 1985.
- ANDERSON, R., “Recent advances in finding best operating conditions”, *J. Amer. Statistic Association.*, 48, 789-798, 1953.
- BISWAS, R.; HAMANN, D.R.. Simulated annealing of silicon atom clusters in langevin molecular dynamics. *Phys. Rev. B*, (34):895–901, 1986.
- CLERC, M., “The swarm and the queen: Toward a deterministic and adaptive particle swarm optimization”, *Proceedings of the 1999 IEEE International Congress on Evolutionary Computation*, Washington, DC, vol. 3, 1951-1957, 1999.
- DEAVEN, D.M.; TIT, N.; MORRIS, J.R.; HO, K.M.. Structural optimization of Lennard-Jones clusters by a genetic algorithm. *Chemical Physics Letters*, (256):195, 1996.
- DOUCET J.P.; WEBER J. *Computer-Aided Molecular Design: Theory and Applications*. Harcour Brace & Company, London, 1996.
- EISENSCHITZ, R.; F. LONDON, Z.; *Physik* 60, 491, 1930.
- FAN, ELLEN. Global Optimization Of Lennard-Jones Atomic Clusters. 2002. 141 f. M Sc, Fevereiro 26, McMaster University, 2002.
- GAINES, L.D.; GADDY, J.L., “Process optimization by flow sheet simulation”, *Ind. Eng Chem. Process Des Dev*, 15, 206-211. 1976.
- GOCKENBACH, M.S.; KEARSLEY, A.J.; SYMES, W.W. An infeasible point method for minimizing the Lennard-Jones potential. *Computational Optimization and Applications*, (8):273– 286, 1997.
- JEZOWSKI J.; BOCHENEK R. J. “Experiences with the use of the Luus-Jaakola algorithm and its modifications in optimization of process engineering problems”, in *Recent developments in optimization and optimal control in chemical engineering*, R. Luus (ed.), Research Signpost, Trivandrum, India, 2002.
- JEZOWSKI, J.; BOCHENEK, R.; ZIOMEK, G., “Random search optimization approach for highly multi-modal nonlinear problems”, *Advances in Engineering Software*, 36, 504-517, 2005.
- KARNOPP, D.C., “Random search techniques for optimization problems”, *Automatica*, 1, 111-121, 1963.
- KIRKPATRICK, S.; GELATT, C.D.; VECCHI, M.. *Science*, (220):671– 680, 1983.

KUBINYI H.; FOLKERS G.; MARTIN Y. C., *3D Qsar in Drug Design: Recent Advances*, New York, 1998.

LAND, A. H.; DOIG, A. G., "An automatic method for solving discrete programming problems," *Econometrica* 28, pp. 497-520, 1960.

LEE, Y.P.; RANGAIAH, G.P.; LUUS, R., "Phase and chemical equilibrium calculations by direct search optimization", *Computers and Chemical Engineering*, 23, 1183-1191, 1999.

LENNARD-JONES, J. E., "On the Determination of Molecular Fields", *Proc. R. Soc. Lond. A* **106** (738): 463-477, 1924.

LIAO, B.; LUUS, R., "Comparison of the Luus-Jaakola optimization procedure and the genetic algorithm", *Engineering Optimization*, 37, 381-398, 2005.

LUUS, R., "Determination of the region sizes for LJ optimization", *Hung. J. Indus. Chem.*, 26, 281-286, 1998.

LUUS, R., "Use of Luus-Jaakola Optimization Procedure for Singular Optimal Control Problems", *nonlinear Analysis*, 47, 5647-5658, 2001.

LUUS, R. e HENNESSY, D., "Optimization of Fed-Batch Reactors by the Luus-Jaakola Optimization Procedure", *Ind. Eng. Chem. Res.*, 38, 1948-1955, 1999.

LUUS, R.; JAAKOLA, T.H.I.; "Optimization by Direct Search and Systematic Reduction of the Size of Search Region", *AIChE Journal*, 19, 760-766, 1973.

MATYAS, J., "Random optimization", *Automation and Remote Control*, 26, 246-253, 1965.

MATSUMOTO, M.; NISHIMURA, T. "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator". *ACM Transactions on Modeling and Computer Simulation* 8 (1): 3-30, 1998.

METROPOLIS N.; ROSENBLUTH A.W.; ROSENBLUTH M.N.; TELLER A.H.; TELLER E.; Equations of state calculations by fast computing machines. *J. Chem. Phys.*, (21):1087-1092, 1958.

NIELSEN, H. B.; "UCTP Test Problems for unconstrained optimization", Technical Report, Department of Mathematical Modelling, University of Denmark, 2000.

PIELA L.; KOSTROWICK J.; SHERAGA H.A. The multiple minima problem in the conformational analysis of molecules: Deformation of the protein energy supersurface by the diffusion equation method. *J. Phy. Chem.*, (93):3339-3346, 1989.

PINCUS, M. A monte carlo method for the approximate solution of certain types of constrained optimization problems. *Oper. Res.*, (18):1225-1228, 1970.

POPLEWSKA, I., PIATKOWSKI, W., ANTOS, D., "Effect of temperature on competitive adsorption of the solute and the organic solvent in reversed-phase liquid chromatography", *Journal of Chromatography A*, 1103, 284-295, 2006.

SACCO, W. F.; ALVES FILHO, H.; PEREIRA, C. M. N. A. Cost-based optimization of a nuclear reactor core design: a preliminary model. 2007 International Nuclear Atlantic Conference, INAC 2007, *Proceedings...*, Santos: ABEN, 2007.

SACCO, W. F.; DE OLIVEIRA, C. R. E. A new stochastic optimization algorithm based on a particle collision metaheuristic. 6th World Congress of Structural and Multidisciplinary Optimization, *Proceedings...*, Rio de Janeiro: WCSMO, 2005.

SACCO, W. F.; DE OLIVEIRA, C. R. E.; PEREIRA, C. M. N. A. Two stochastic optimization algorithms applied to nuclear reactor core design. *Progress in Nuclear Energy*. Vol. 48, pp. 525-539, 2006.

SACCO, W. F.; LAPA, C. M. F.; PEREIRA, C. M. N. A.; ALVES FILHO, H. A metropolis algorithm applied to a nuclear Power plant auxiliary feedwater system surveillance tests policy optimization. *Progress in Nuclear Energy*. N. 50, pp. 15-21, 2008.

SALCEDO, R.L.; GONÇALVES, M.J.; AZEVEDO, S.F. *An improved random search algorithm for non-linear optimization*, Computers and Chemical Engineering, 14, 1111-1126, 1990.

SHALLOWAY, D. Packet annealing: A deterministic method for global minimization, application to molecular conformation. In *Recent Advances in Global Optimization*, pages 433–477. Princeton University Press, 1992.

SINGH, V. “Obtaining Routh-Padé approximants using the Luus-Jaakola algorithm”, IEE Proc.-Control Theory Appl., 152, 129-132, 2005.

STORN, R.; PRICE, K. "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces". *Journal of Global Optimization* 11: 341–359, 1997.

WALES D.J.; DOYE J.P.K.. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *Chem. Phys. Lett.*, (269):408–412, 1997.

WILLE L.T. Minimum-energy configurations of atomic clusters: New results obtained by simulated annealing. *Chem. Phys. Lett.*, (133):405–410, 1987.

WILLE L.T.; VENNIK J. Computational complexity of the ground-state determination of atomic clusters. *J. Phys.*, A(18):L419, 1985.

WOLPERT, D. H.; MACREADY, W. G., “No Free Lunch Theorems for Optimization”, IEEE Trans. Evol. Conq., Vol 1, pp. 67-82, 1997.