

Aluno: David Dobrigkeit Chinellato  
Orientador: Douglas S. Galvão

**Dinâmica Molecular de Nanoosciladores  
Mecânicos Baseados em Nanotubos de Carbono**

*Relatório parcial de Progresso*

*Referente ao Período de Fevereiro até Julho de 2003*

*Processo Número: 12182-1*

08 de Julho de 2003  
UNICAMP, Campinas, SP

## **O projeto: Recapitulação**

Recentemente, tem havido crescente interesse em estruturas de escala nanométrica. Especificamente, nanotubos baseados em carbono têm despertado grande interesse por parte de pesquisadores.

Um trabalho experimental de grande relevância neste contexto é o de Cummings e Zettl [1], no qual foi demonstrado que dispositivos baseados em nanotubos multicamadas podem atuar como extensões telescópicas. Além disso, tais sistemas poderiam eventualmente ser usados como nanoosciladores, utilizando a atração de Van der Waals como força restauradora. Porém, seria interessante determinar sobre quais circunstâncias o oscilador permanece razoavelmente estável – especificamente, considerações como temperatura e flutuações sistêmicas.

### O cronograma

Na proposta inicial, estipulamos como meta semestral a revisão bibliográfica, a melhor compreensão dos métodos associados à dinâmica molecular, e familiarização com a metodologia utilizada, incluindo a linguagem de programação FORTRAN. No semestre seguinte, abordaríamos o problema em questão, o dos nanoosciladores, e obteríamos os resultados necessários.

Uma revisão rápida das nossas atividades sugere que cumprimos tal objetivo. Como primeira etapa de aprendizado, começando a partir da etapa mais básica da metodologia, estudamos um algoritmo de enrolamento de nanotubos – um artifício matemático necessário para a construção das estruturas utilizadas na dinâmica molecular. Adicionalmente, já modificamos o programa adicional para diferentes motivos estruturais, como o grafino e a sua variante consideravelmente mais complexa, o assim chamado grafino supramolecular. Estas modificações indicam a compreensão obtida durante as fases iniciais de estudo do algoritmo.

Depois desta etapa inicial, estudamos os métodos e artifícios associados à dinâmica molecular, como campos de força, minimizadores de energia e algoritmos de simulação diversos. Apenas após adequados estudos destes métodos partimos para um problema inicial, bastante semelhante ao nosso objetivo final, o dos nanoosciladores baseados em sistemas tubo-tubo.

O primeiro problema considerado, cujo propósito era principalmente o de treinamento, era o de encapsulamento de moléculas de C<sub>60</sub> dentro de nanotubos de diâmetro apropriado, como no trabalho de Berber e Kwon [2]. Encontramos, através de diversas simulações, os menores tubos para os quais o encapsulamento ocorre; além disso, estudamos também o comportamento das moléculas de C<sub>60</sub>, uma vez encapsuladas. Constatamos uma oscilação pouco estável para uma molécula isolada dentro do tubo, com frequência inicial de cerca de 8Ghz. Encapsulamentos de múltiplas moléculas causam a formação de estruturas conhecidas como “peapods”, nas quais vários C<sub>60</sub>s ficam agrupados dentro do tubo. Confirmamos também a formação destas estruturas, e constatamos propriedades mecânicas semelhantes às medidas experimentalmente por Hornbaker et al [3].

A grande familiaridade com a metodologia utilizada em simulações de dinâmica molecular que foi conseqüência deste primeiro estudo de treinamento será utilizada, a seguir, para uma investigação do cenário nanooscilador tubo-tubo. Ambos os problemas são de fronteira e são de grande interesse científico.

Este relatório também inclui um CD contendo dois filmes de simulações de dinâmica molecular do encapsulamento, ambos obtidos com os métodos descritos aqui e o auxílio dos pacotes computacionais incluídos no Cerius2 [4].

# Índice

## 1 – Os Nanotubos

1.1 Introdução	5
1.2 Como são gerados os Nanotubos	5
1.3 As estruturas do Tubo - Conclusão	14
1.4 Nanotubos baseados em diferentes variantes estruturais	16
1.5 Diâmetros de diferentes Nanotubos	17

## 2 – As Simulações

2.1 Introdução	18
2.2 O campo de força	20
2.3 O campo de força expressado matematicamente	21
2.4 Integração da equação de Newton	25
2.5 Os ensembles estatísticos	28
2.6 Os Minimizadores de Energia	32
2.7 O encapsulamento de fulerenos	35
2.8 Análise dos resultados e conclusão	42

Bibliografia	44
--------------	----

## 1.1 Introdução

Temos como intuito simular o processo de encapsulamento do C60 em nanotubos de tamanho apropriado e investigar em quais situações o encapsulamento ocorre. Este será a etapa inicial de treinamento. Posteriormente, vamos investigar também a formação de estruturas conhecidas como “peapods”, nas quais várias moléculas de C60 se encontram encapsuladas dentro de um nanotubo. Depois, pretendemos utilizar o conhecimento adquirido durante a fase de treinamento e reproduzir o resultado dos nanoosciladores baseados em nanotubos oscilando dentro de nanotubos.

Primeiramente, é importante deixarmos claro que estamos lidando com simulações que, pelo menos em primeiro plano, já foram pré-programadas nos programas utilizados. Neste sentido, é importante que seja feita uma pesquisa adequada do que está sendo usado, pois o uso adequado dos programas depende em grande parte da boa compreensão da teoria utilizada pelo mesmo.

Inicialmente, vamos explicar o funcionamento do algoritmo gerador de nanotubos e demonstrar uma pequena variação do mesmo, que poderá ser utilizada mais tarde – especificamente, uma modificação que permite gerar um nanotubo com o motivo estrutural de uma variante do grafino.

Pretendemos então estudar a metodologia envolvida nas simulações, conhecimento este necessário para uma boa avaliação dos resultados e para o processo de simulação em si. Depois, vamos simular o encapsulamento dos fulerenos (neste caso, moléculas de C60) e a sua conformação dentro dos nanotubos capazes de contê-los. Pretendemos então comparar o sistema de fulerenos encapsulados com sistemas de nanotubos multicamadas – especificamente, nanoosciladores tubo-tubo.

## 1.2 Como são gerados os Nanotubos?

Os programas de simulação, embora oferecendo vários recursos, não têm geradores de estruturas mais sofisticadas, mas admitem vários formatos básicos (tabelas ASCII

dispostas de maneira convenientes) para o posicionamento e o tipo de átomos. Para que possamos adequadamente simular os nanotubos, precisamos portanto de uma maneira eficiente e genérica para gerar a estrutura enrolada dos nanotubos. Primeiramente, vamos ver qual é a idéia por trás de um algoritmo capaz de, dadas as características de um tubo pelo usuário, gerar as posições dos átomos deste tubo.

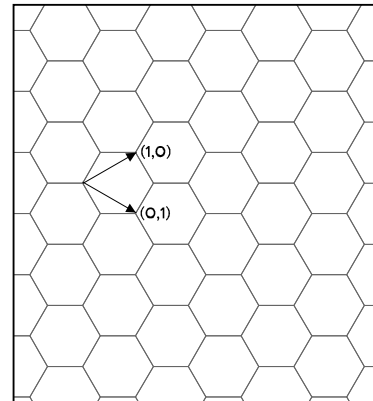


Figura 1: O plano a ser enrolado.

Os tubos são basicamente enrolamentos específicos de um plano composto por hexágonos – uma colméia, por assim dizer – e existem muitas combinações possíveis de enrolamento. Na figura 1, vemos o plano de hexágonos a ser enrolado e uma base para este plano, já convenientemente escolhida de tal maneira que qualquer combinação linear destes dois vetores seja um vértice de algum hexágono. Mais especificamente, vamos usar esta base para definir um vetor – que chamaremos de vetor *quiral*  $\mathbf{C}$  – que será dobrado de maneira tal que ele se torne a circunferência de um nanotubo. Desta forma, o módulo do vetor quiral será o perímetro do nanotubo gerado. Este vetor quiral será dado pelo usuário na forma  $(n,m)$ , com  $n$  e  $m$  inteiros, nas coordenadas da base especificada na figura 1, ou seja,  $\mathbf{C} = n\mathbf{a}_1 + m\mathbf{a}_2$ , se chamarmos o vetor  $(1,0)$  de  $\mathbf{a}_1$  e o vetor  $(0,1)$  de  $\mathbf{a}_2$ . Alguns autores também definem o primeiro vetor da base de maneira trocada em relação à vista aqui; teoricamente, porém, isso não é relevante. Trocar as coordenadas neste caso equivale a um espelhamento na horizontal da figura 1, e o plano é invariante a este espelhamento (i.e. continuaremos com a mesma estrutura hexagonal, sem modificações). Note que o plano tem dimensões já definidas, pois as ligações C-C, que são os lados dos hexágonos, estão fixas em  $1,42\text{Å}$ . Desta forma, os módulos de  $\mathbf{a}_1$  e  $\mathbf{a}_2$  estão ambos fixos em  $3^{(1/2)}(1,42\text{Å})$ , que chamaremos simplesmente de “ $a$ ”.

Em relação aos valores  $n$  e  $m$  dados pelo usuário, algumas denominações são muito comuns na literatura: aos tubos com  $n = m$  damos o nome de tubos *Armchair*. Além disso, aos tubos com  $m = 0$  ou  $n = 0$ , damos o nome de *Zigzag*. E aos outros tubos, onde  $nm \neq 0$  e

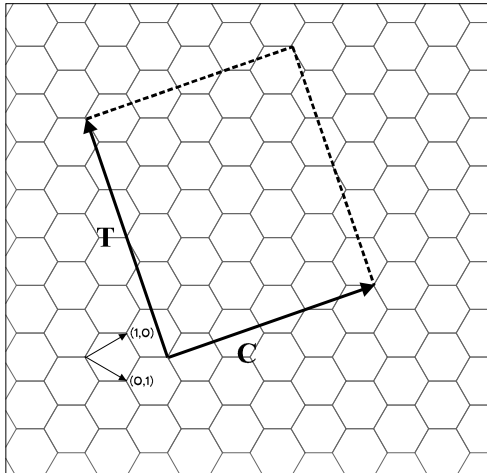


Figura 3: A célula unitária, desenrolada.

$n \neq m$ , damos o nome de tubos quirais. Os dois primeiros têm estes nomes devido aos padrões vistos ao longo do perímetro do nanotubo, que podem ser vistos na figura 2. Em vermelho estão representados os padrões *zigzag*, e em azul, os padrões *armchair*.

Seria então conveniente encontrarmos uma maneira de criar uma célula unitária do tubo – um padrão que se repete. Para tanto, precisaremos achar um vetor especial, que chamaremos de **T**, que deve ter propriedades específicas: Este vetor

deve ser a menor combinação linear dos dois vetores da base com coeficientes inteiros que ainda é perpendicular ao vetor quiral **C** especificado. O vetor **T** pode ser visto na figura 2,

onde também podemos ver que o quadrado de lados **C** e **T** é convenientemente repetido ao longo do motivo estrutural hexagonal. Essencialmente, este quadrado deverá ser mapeado para o manto (o lado) de um cilindro que chamaremos de *cilindro unitário* do nanotubo – justamente pela propriedade de simetria deste quadrado  $C \times T$ . Para um tubo excepcionalmente longo, bastará então empilhar cilindros unitários e o tubo estará automaticamente encaixado. Dados  $(n, m)$ , precisamos então apenas encontrar as coordenadas de **T**  $(t_1(n, m), t_2(n, m))$ . Para tanto, vamos inicialmente investigar o funcionamento do produto escalar na

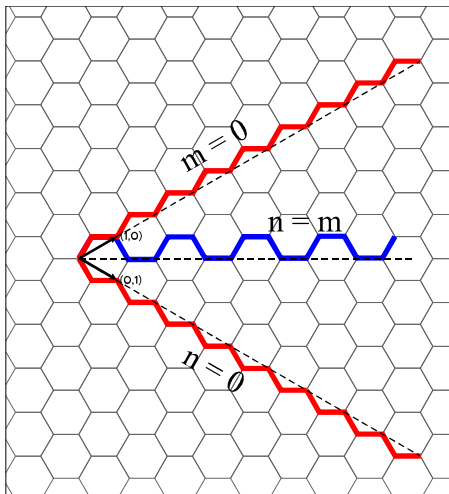


Figura 2: Os tubos Zigzag e Armchair

base  $\{\mathbf{a}_1, \mathbf{a}_2\}$ ; depois, basta encontrar um vetor qualquer sempre perpendicular ao vetor **C** e reduzi-lo ao menor tamanho ainda inteiro, dividindo-o pelo maior divisor comum de suas coordenadas:

<p><i>Propriedades básicas:</i></p> $\langle \mathbf{a}_1, \mathbf{a}_1 \rangle = \langle \mathbf{a}_2, \mathbf{a}_2 \rangle = a^2$ $\langle \mathbf{a}_1, \mathbf{a}_2 \rangle = \langle \mathbf{a}_2, \mathbf{a}_1 \rangle = a^2/2$ $\langle (x\mathbf{a}_1 + y\mathbf{a}_2), (x\mathbf{a}_1 + y\mathbf{a}_2) \rangle = a^2(x^2 + y^2 + xy)$	$\langle \mathbf{C}, \mathbf{T} \rangle = 0$ é simplesmente... $\langle (n\mathbf{a}_1 + m\mathbf{a}_2), (t_1\mathbf{a}_1 + t_2\mathbf{a}_2) \rangle = 0$ $a^2(1/2)[t_1(2n + m) + t_2(2m + n)] = 0$
--	---

Não é difícil ver que o vetor  $(t_1, t_2) = (2m + n, -2n - m)$  sempre obedece a relação acima. Não há nenhuma garantia, porém, de que este vetor seja de fato o menor vetor de coeficientes inteiros – é possível que ele seja um múltiplo deste menor vetor, ou seja, que tenhamos encontrado  $(\alpha t_1, \alpha t_2)$ . Como  $\mathbf{T}$  deve ser o menor vetor de coeficientes ainda inteiros, é intuitivo que  $t_1$  e  $t_2$  sejam primos entre si e  $\alpha$  é, de fato, o maior divisor comum (MDC) dos valores obtidos,  $(2m + n, -2n - m)$ . Vamos escrever o MDC como sendo um operador nos parágrafos seguintes, por ser conveniente, da forma  $\text{mdc}(\alpha, \beta) \equiv$  “maior divisor comum de  $\alpha$  e  $\beta$ ”.

Existe uma propriedade que relaciona o  $\text{mdc}(n, m)$  com  $\text{mdc}[(2m+n), (2n+m)] (= \text{mdc}[(2m+n), (-2n-m)])$  : **Se** o  $\text{mdc}(n, m)$  for divisor de  $n-m$ , **então** o  $\text{mdc}[(2m+n), (2n+m)] = 3\text{mdc}(n, m)$ . **Caso contrário**,  $\text{mdc}[(2m+n), (2n+m)] = \text{mdc}(n, m)$ . Esta propriedade matemática pode ser provada, mas omitimos a prova aqui. Evidentemente, tratar-se-á de uma simplificação na programação mais tarde, mas não haveria nenhum problema em simplesmente calcular o MDC de  $(2m+n)$  e  $(2n+m)$  diretamente no lugar do MDC de  $n$  e  $m$ .  
 Concluindo:

$$t_1(n, m) = (2m + n)[\text{mdc}(2m+n, 2n+m)]^{-1} \quad (I)$$

$$t_2(n, m) = (-2n - m)[\text{mdc}(2m+n, 2n+m)]^{-1} \quad (II)$$

Resolvido o problema de achar  $t_1$  e  $t_2$ , temos agora o nosso quadrado a ser enrolado. Note que, enquanto o módulo de  $\mathbf{C}$  era igual ao perímetro da circunferência-base do cilindro unitário, o módulo de  $\mathbf{T}$  nada mais é do que a altura deste mesmo cilindro.

O problema agora fica sendo mapear as posições dos átomos individualmente neste quadrado. Note que, se escrevermos a posição dos átomos como sendo  $(\gamma \mathbf{C} + \lambda \mathbf{T})$ , com  $\gamma < 1$  e  $\lambda < 1$ , podemos dizer que o átomo será posicionado, em coordenadas cilíndricas



$(r, \theta, z)$ , nas coordenadas  $(|C|/2\pi, 2\pi y/\lambda|T|)$ , como foi tudo convenientemente escolhido. E é este mapeamento que está no cerne do programa que será utilizado.

Achar as posições dos átomos individualmente é um problema interessante de teoria de grupos, que pode ser abordado de maneira mais rigorosa do que a pretendida aqui. Alguns resultados interessantes saem de uma abordagem mais rigorosa; alguns destes serão adicionados posteriormente.

O problema que precisamos resolver é, essencialmente, um problema de simetria. Temos um plano de hexágonos e precisamos gerá-lo de maneira conveniente; para tanto, precisamos saber o que controla a periodicidade do mesmo. Note que, para cada hexágono do plano, basta colocarmos dois vértices e teremos gerado a estrutura toda depois de um número suficiente de repetições; de fato, estes vértices estão contidos no que chamaremos de *célula unitária* (com a ressalva de não trocar esta célula unitária pela *outra* “célula unitária”, proveniente do enrolamento do nanotubo, que chamamos de cilindro unitário por causa desta razão) em relação à simetria do plano hexagonal.

Portanto, o nosso programa deverá localizar cada célula unitária contida dentro do quadrado  $C \times T$  e posicionar dois átomos dentro de cada uma destas células de repetição. A área ocupada por uma destas células é idêntica à área ocupada por um hexágono; então, se dividirmos  $|C||T|$  pela área de 1 hexágono, teremos metade do número de átomos dentro do cilindro unitário. A área de um hexágono é simplesmente  $|a_1 \times a_2|$ , onde “x” denota o produto vetorial entre  $a_1$  e  $a_2$ , e pode ser escrita como  $3(3)^{(1/3)}a^2/2$ . Este resultado fica, depois de um pouco de álgebra, sendo:

$$N = |C||T|/3(3)^{(1/3)}a^2/2 =$$

$$2(n^2+m^2+nm)/\text{mdc}(2m+n, 2n+m) \text{ (III)}$$

número de átomos no cilindro unitário:

$$4(n^2+m^2+nm)/\text{mdc}(2m+n, 2n+m)$$

O que precisamos, portanto, é determinar as posições das células unitárias, de acordo com a figura

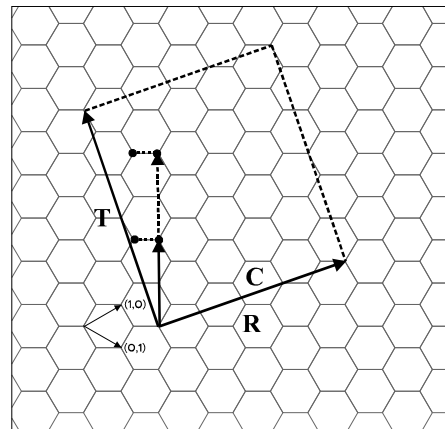


Figura 4: O vetor  $R$  e os átomos A e B.

4. Usaremos o vetor de simetria  $\mathbf{R}$ , como na figura, da seguinte maneira: A primeira célula unitária está posicionada na origem dos vetores  $\mathbf{C}$  e  $\mathbf{T}$ . As células unitárias seguintes são encontradas somando-se múltiplos de  $\mathbf{R}$  (mais especificamente, serão  $\mathbf{N}-1$  múltiplos) à posição original,  $\mathbf{O}$ , sempre deslocando a posição atual para que esta esteja contida dentro do cilindro unitário. Vemos na imagem também os dois átomos – que chamamos de A e B – que constituem a célula unitária que deverá ser repetida. Cada átomo da célula unitária deve ser incluído separadamente.

Para achar o vetor  $\mathbf{R}$ , vamos ver as propriedades que ele satisfaz: de todos os vetores-posição de átomos dentro do quadrado unitário, ele é aquele que tem a menor projeção não-nula em  $\mathbf{C}$  (já que nós já introduzimos uma parametrização anteriormente: é a posição daquele átomo com menor  $\gamma$ ). Encontrá-lo é o principal problema do programa todo, já que, dada a posição de um dos átomos da célula unitária, podemos usar o ângulo entre o vetor quiral  $\mathbf{C}$  e um dos vetores da base – digamos  $a_1$  – para posicionar os outros átomos da célula unitária depois de uma rotação apropriada, usando praticamente o que é um sistema de coordenadas polares ao redor da posição do átomo A. Mas ainda não chegamos lá. Antes disso, é necessário achar  $\mathbf{R} \equiv (p(n,m), q(n,m))$ .

Já que dispomos de um computador e estamos programando, vamos testar todos os vetores possíveis contidos dentro de  $\mathbf{C} \times \mathbf{T}$ . Para tanto, pretendemos experimentar valores  $p$ ,  $q$  tal que  $-t_1 \leq p \leq t_1$  e  $-t_2 \leq q \leq t_2$ . Deve existir apenas um vetor  $\mathbf{R}$  que satisfaz as nossas necessidades, e podemos checar todos, convenientemente, na possibilidade de existirem dois (ou nenhum) dentro do intervalo checado. As condições que  $\mathbf{R}$  satisfaz são as seguintes:

1.  $t_1 q - t_2 p = 1$  (IV)
2.  $mp - nq > 0$  (V)
3.  $mp - nq < \mathbf{N}$  (VI)

Estas condições não são muito intuitivas à primeira vista. A primeira condição basicamente reside no fato de que o vetor tem a menor componente possível em  $\mathbf{C}$ . Dado que o vetor varrerá o espaço total do quadrado, deve valer que a área varrida em um

retângulo de base  $p$  paralela ao vetor  $\mathbf{C}$  e de altura dada pelo  $|\mathbf{T}|$  deve ser igual à área de um hexágono:

$$\begin{aligned} \mathbf{a}_1 \times \mathbf{a}_2 &= \mathbf{T} \times \mathbf{R} \\ \mathbf{a}_1 \times \mathbf{a}_2 &= (t_1 \mathbf{a}_1 + t_2 \mathbf{a}_2) \times (p \mathbf{a}_1 + q \mathbf{a}_2) = t_1 q (\mathbf{a}_1 \times \mathbf{a}_2) + t_2 p (\mathbf{a}_2 \times \mathbf{a}_1) = (\mathbf{a}_1 \times \mathbf{a}_2) (t_1 q - t_2 p) \\ \therefore (t_1 q - t_2 p) &= 1 \end{aligned}$$

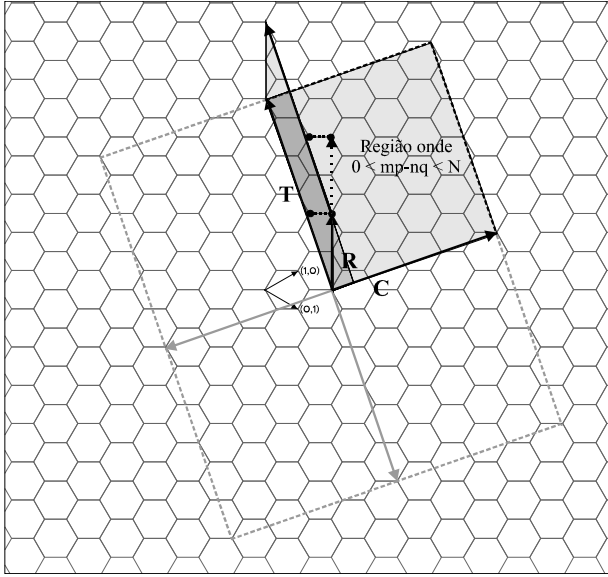


Figura 5: As condições para  $\mathbf{R}$  ser encontrado.

A segunda condição garante que o vetor  $\mathbf{R}$  está simplesmente contido no cilindro unitário, e não nos cilindros unitários vizinhos; a terceira condição garante basicamente a mesma coisa. A figura 5 mostra o quadrado que contém todos os  $\mathbf{R}$ s que o programa avaliará.

Encontrado  $\mathbf{R}$  – e apenas um  $\mathbf{R}$  – podemos encontrar todas as células unitárias dentro do cilindro unitário e posicionar os átomos. Na figura 3, demos ao primeiro átomo a ser posicionado a denominação “Átomo A”. De fato, este átomo é facilmente encontrado através de  $\mathbf{O} + i \mathbf{R}$ , onde  $\mathbf{O}$  é a origem do sistema, de onde saem os vetores  $\mathbf{T}$  e  $\mathbf{C}$ , com  $0 < i < N-1$ , o que resulta em  $N$  posições.

Antes de posicionar o átomo B, vamos encontrar alguns ângulos convenientes para que a projeção das posições atômicas sobre  $\mathbf{C}$  e  $\mathbf{N}$  seja facilitada posteriormente. Primeiramente, vamos encontrar o que chamaremos de ângulo quiral  $\varphi$ , o ângulo entre  $\mathbf{C}$  e  $\mathbf{a}_1$ , assim como o ângulo  $\psi$  entre  $\mathbf{R}$  e  $\mathbf{a}_1$ . Estes dois ângulos são necessários para encontrar o ângulo entre  $\mathbf{C}$  e  $\mathbf{R}$ ,  $\xi$ , que é vital para a projeção das posições atômicas. Para conseguir isto, basta usarmos as definições de seno e cosseno convenientemente:

$$\begin{aligned} \text{Tg}(\varphi) &= \text{Sin}(\varphi)/\text{Cos}(\varphi) = ((n \mathbf{a}_1 + m \mathbf{a}_2) \times \mathbf{a}_1) / \langle n \mathbf{a}_1 + m \mathbf{a}_2, \mathbf{a}_1 \rangle \\ &= 3^{(1/3)} m / (2n+m) \rightarrow \varphi = \text{ArcTg}[3^{(1/3)} m / (2n+m)] \end{aligned}$$

Note que o ângulo  $\psi$  pode ser obtido de maneira inteiramente análoga. As duas relações, assim como a expressão (bastante simples) de  $\xi = \varphi - \psi$ , estão escritas abaixo e indexadas com (VII), (VIII) e (IV). Estes ângulos são suficientes na determinação da posição do átomo A; porém, temos um outro átomo ainda na célula unitária.

A posição do átomo B pode ser encontrada simplesmente considerando um vetor pequeno, que chamaremos de **AB** e que cobre o lado do hexágono, ligando o átomo A ao B, conforme a linha pontilhada na figura 3 já sugere. Note que o ângulo entre  $\mathbf{a}_1$  e a horizontal é de  $30^\circ$ . Dados (n,m), fica simples encontrar **AB**, bastando encontrar o ângulo entre o vetor  $\mathbf{a}_1$  e o vetor quiral e efetuando as projeções, com um pouco de trigonometria, e teremos as relações (XI) e (XII).

ângulo entre <b>C</b> e $\mathbf{a}_1$ :	$\varphi = \text{ArcTg}[3^{(1/3)}m/(2n+m)]$ (VII)
ângulo entre <b>R</b> e $\mathbf{a}_1$ :	$\psi = \text{ArcTg}[3^{(1/3)}q/(2p+q)]$ (VIII)
ângulo entre <b>C</b> e <b>R</b> :	$\xi = \varphi - \psi$ (IX)
ângulo de variação, cada átomo A	$\zeta = 2\pi/N$ (X)
diferença do ângulo, átomo A e B	$\varepsilon = (1.42\text{\AA}/ C ) \text{Cos}(\pi/6 - \varphi)/(2\pi)$ (XI)
diferença em z, átomo A e B	$\Delta z = (1.42\text{\AA}) \text{Sin}(\pi/6 - \varphi)$ (XII)

Para finalizar, basta lembrar que usaremos um mapeamento cilíndrico da forma  $(|C|/2\pi, 2\pi\gamma, \lambda T)$  para cada um dos átomos posicionados em  $(\gamma \mathbf{C} + \lambda \mathbf{T})$ , e acharemos  $\gamma$  e  $\lambda$  utilizando inicialmente as relações de (I) até (VI), que explicitam todos os vetores relevantes na base  $\{\mathbf{a}_1, \mathbf{a}_2\}$ , e posteriormente as relações de (VII) até (XII), que fornecem todos os ângulos necessários para a projeção das posições atômicas sobre **C** e **T**.

Resumindo o algoritmo do programa, o usuário vai fornecer apenas (n,m) e o programa deve gerar uma tabela em um formato ASCII especificamente espaçado e

seguindo os detalhes necessários para servir de entrada para um programa de simulações adequado. Vamos omitir estes detalhes, devido à sua pouca importância no contexto geral.

O programa segue, portanto, as seguintes regras:

1. Usuário entra com (n,m)
2. Encontra-se  $t_1$  e  $t_2$  de acordo com (I) e (II). Além disso, encontra-se o raio do nanotubo, que vamos chamar de  $\rho$  (dado por  $|\mathbf{C}|/2\pi = a^2(n^2+m^2+nm)/2\pi$ ), o ângulo  $\phi$  entre a horizontal e  $\mathbf{C}$ , o ângulo  $\psi$  entre  $\mathbf{R}$  e  $\mathbf{C}$  e o número de células unitárias dentro do cilindro unitário,  $N$ , de acordo com (III)
3. Encontra-se o vetor de simetria,  $\mathbf{R}$ , de acordo com as condições (IV), (V) e (VI), testando todas as possibilidades  $\mathbf{R} \equiv (p,q)$  com  $-t_1 \leq p \leq t_1$  e  $-t_2 \leq q \leq t_2$ .
4. Considerando  $\mathbf{A}_i \equiv (x_i, y_i, z_i)$ , com  $0 < i < N-1$ , como sendo as  $N$  posições atômicas indexadas, vamos considerar o seguinte procedimento: Note que, para o termo  $z$ , usamos um operador “parte inteira de”, equivalente a truncar um número em um computador, e denotamos este operador de  $[\lambda]$ . O que precisamos aqui é simplesmente achar a parte decimal de  $\lambda$ , já que  $\mathbf{O} + i \mathbf{R}$  sai do quadrado  $\mathbf{C} \times \mathbf{T}$  alguma hora e deve entrar no quadrado seguinte (onde a projeção em  $\mathbf{T}$  varia entre 1 e 2, e não 0 e 1). Portanto, vamos usar o nosso operador  $[\lambda]$  para encontrar a parte inteira de  $\lambda$  e tirá-la de  $\lambda$ , ou seja,  $(\lambda - [\lambda])$ , e teremos a parte decimal de  $\lambda$ . Esta parte nos dá a posição do átomo *recolocado dentro* do quadrado  $\mathbf{C} \times \mathbf{T}$ , dada a periodicidade deste quadrado. As relações abaixo resumem isto:

Para  $i$  ímpar: Átomo  $A$  (de índice  $i$ )

$$x_i = \rho \cos[\zeta_i]$$

$$y_i = \rho \sin[\zeta_i]$$

$$z_i = \left( i|\mathbf{R}| - \left[ \left( \frac{i|\mathbf{R}|\sin(\xi)}{|\mathbf{T}|} \right) \right] \left( \frac{|\mathbf{T}|}{\sin(\xi)} \right) \right) \sin(\xi)$$

$$\text{Relações auxiliares: } \left\{ \begin{array}{l} \lambda = \frac{(i|R|)}{|T|} \sin(\xi) \\ \lfloor \lambda \rfloor = \left\lfloor \left( \frac{(i|R|)\sin(\xi)}{|T|} \right) \right\rfloor \end{array} \right.$$

Para  $i$  par: Átomo  $B$  (de índice  $i$ )

$$x_i = \rho \cos[\zeta_i + \varepsilon]$$

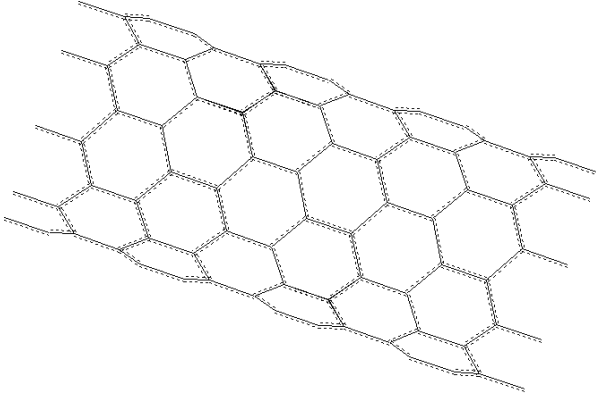
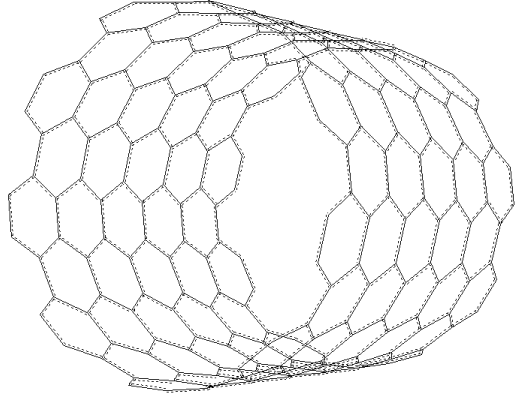
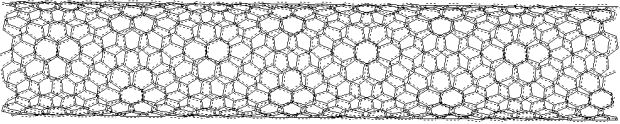
$$y_i = \rho \sin[\zeta_i + \varepsilon]$$

$$z_i = \left( i|R| - \left\lfloor \left( \frac{(i|R|)\sin(\xi)}{|T|} \right) \right\rfloor \left( \frac{|T|}{\sin(\xi)} \right) \right) \sin(\xi) - \Delta z$$

De qualquer forma, para cada átomo devemos checar se a coordenada  $z$  excede  $|T|$  - se ela o fizer, o átomo está fora do cilindro unitário e deve ser deslocado em  $z$  por um múltiplo inteiro de  $|T|$ , até que  $0 < z_i < |T|$ . Convenientemente, não precisamos checar se o ângulo  $(\zeta_i + \varepsilon)$ , que nada mais é do que  $\theta$  em coordenadas cilíndricas, está entre  $0$  e  $2\pi$ , pois se  $\theta$  exceder  $2\pi$ , a posição do átomo já estará correta de novo e não precisamos deslocar ela - as funções trigonométricas são todas periódicas, afinal.

### 1.3 As estruturas dos tubos - conclusão

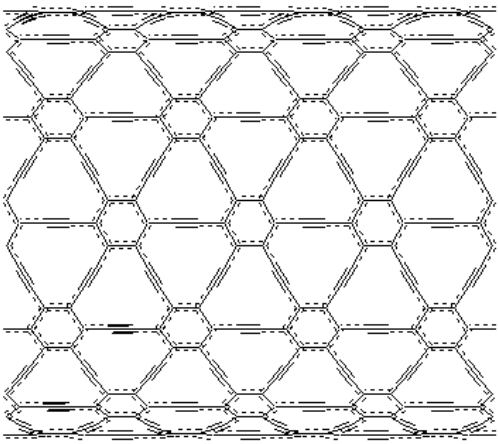
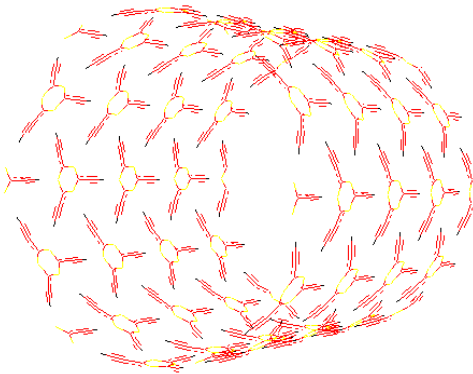
O programa visto aqui dá grande liberdade ao seu usuário, já que é capaz de gerar qualquer enrolamento  $(n,m)$ . Alguns exemplos se encontram abaixo, gerados com o próprio programa e visualizados com um programa simples de simulações. Consideramos, de acordo com a nomenclatura sugerida na figura 2, três exemplos: um tubo *zigzag*, um tubo *armchair* e um tubo quiral (especificamente,  $(9,0)$ ,  $(9,9)$  e  $(11,7)$ ).

	<p><u>Tubo ZigZag</u>  Entrada do usuário: (9,0)  Número de cilindros unitários empilhados: 4</p>
	<p><u>Tubo Armchair</u>  Entrada do usuário: (9,9)  Número de cilindros unitários empilhados: 6</p>
	<p><u>Tubo Quiral</u>  Entrada do usuário: (11,7)  Número de cilindros unitários empilhados: 1*</p>

\* - note como o cilindro unitário de um tubo quiral é consideravelmente maior do que o cilindro unitário para os tubos *armchair* e *zigzag*. Isto já era de se esperar; quando o tubo é *armchair* ou *zigzag*, o vetor **T** não é necessariamente comprido e as relações (I) e (II) são muito simplificadas. Mas genericamente, o tubo quiral é muito mais sofisticado e **T** pode ser surpreendentemente comprido. Exemplificamos este fato de maneira especialmente visível, já que 11 e 7 são ambos primos e o tubo (11,7) é de fato um tubo de cilindro unitário excepcionalmente comprido, mesmo para um tubo quiral.

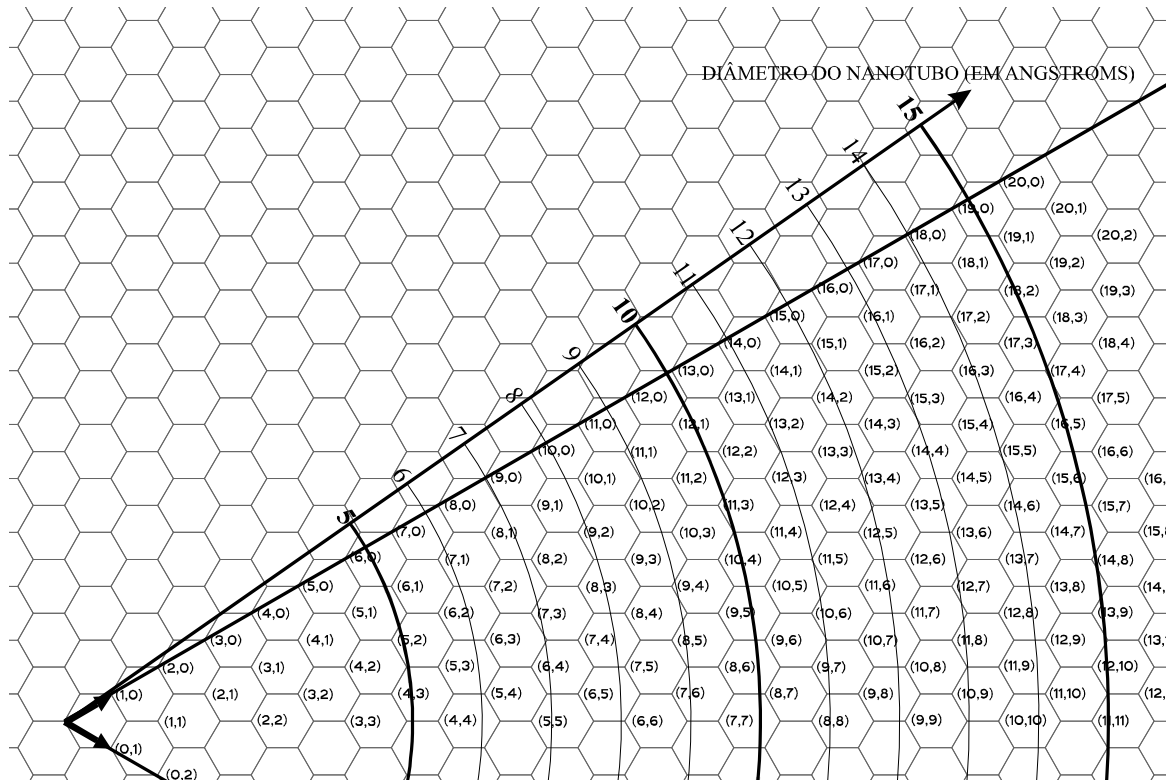
## 1.4 Nanotubos baseados em variantes estruturais

Com uma pequena variação do algoritmo explicado aqui, fomos capazes de criar nanotubos com motivos estruturais diferentes, como os vistos abaixo. O vetor de simetria  $\mathbf{R}$  é suficiente genérico para servir para qualquer simetria  $D_3$ , e basta criar um algoritmo que gere a célula de simetria nos locais especificados pelos múltiplos inteiros de  $\mathbf{R}$ , considerando a célula já apropriadamente rotacionada. Fazendo exatamente isto, fomos capazes de gerar algumas variantes estruturais para as quais não tínhamos programas originalmente.

	<p><u><i>Nanotubo baseado no Grafino</i></u></p> <p>Esta foi a primeira variante de nanotubos que o programa modificado gerou. A célula unitária consiste de</p> <p>12 átomos, todos de carbono. Este é um tubo teórico e não foi obtido experimentalmente em lugar algum. É estável, ao contrário do tubo abaixo.</p>
	<p><u><i>Nanotubo baseado no grafeno supramolecular</i></u></p> <p>Motivo estrutural consideravelmente diferente, com célula unitária de 15 átomos, com 9 de carbono, três de hidrogênio e três de nitrogênio. Simulações de dinâmica molecular sugerem um sistema instável, devido à ausência de ligações químicas – a interação é predominantemente de pontes de hidrogênio entre cada uma das moléculas com formato de Y.</p>



## 1.5 Figura: Diâmetros de diferentes Nanotubos



Note que os arcos representam os diâmetro (em Å) dos tubos gerados. Consultando este desenho, é possível achar rapidamente tubos de diâmetro entre 5 Å a 10 Å.

## 2. Introdução à dinâmica molecular

### 2.1 - O que é dinâmica molecular

Vamos agora introduzir os métodos comumente usados para analisar o comportamento de sistemas de escala molecular. A descrição completa do sistema, considerando efeitos quânticos e relativísticos, está muito além das nossas necessidades neste momento e é um problema bastante sofisticado. Por esta razão, vamos começar a nossa análise com a equação de Schrödinger não-relativística e independente do tempo: (note que os vetores  $\mathbf{R}$  e  $\mathbf{r}$  devem conter informações a respeito do sistema todo, ou seja, todos os vetores posição  $\mathbf{R}_i$  – portanto,  $\mathbf{R}$  pode ser tratado matematicamente como uma matriz. O mesmo vale para  $\mathbf{r}$ ).

$$H\psi(\vec{R}, \vec{r}) = E\psi(\vec{R}, \vec{r}) \quad (2.1.1)$$

Na equação 2.1.1,  $H$  representa o operador Hamiltoniano para este sistema,  $\psi$  é a função de onda e  $E$  é a energia (autovalor associado à autofunção  $\psi(\mathbf{R}, \mathbf{r})$ ). As coordenadas relevantes para o problema incluem  $\mathbf{R}$ , as posições dos núcleos atômicos, e  $\mathbf{r}$ , as posições dos elétrons associados.

Resolver a equação 2.1.1 ainda é um procedimento muito intrincado para ser prático. Para ser de algum uso, vamos tomar algumas aproximações na equação 2.1.1. Em particular, Born e Oppenheimer sugeriram, em 1927, que seria possível desacoplar o movimento dos elétrons dos núcleos em duas equações diferenciais separadas, considerando que o núcleo dos átomos é pelo menos três ordens de grandeza mais pesado do que os elétrons. As equações separadas, 2.1.2 e 2.1.3, são consideravelmente mais práticas.

$$\begin{cases} H\psi(\vec{r};\vec{R}) = E\psi(\vec{r};\vec{R}) & (2.1.2) \\ H\phi(\vec{R}) = E\phi(\vec{R}) & (2.1.3) \end{cases}$$

Note que, na equação 2.1.2, a posição nuclear  $\mathbf{R}$  é apenas um parâmetro constante. A equação 2.1.2 define uma energia  $E(\mathbf{R})$ , função apenas das coordenadas nucleares. A esta energia damos o nome de *superfície de energia potencial*; desta forma, a equação 2.1.3 definirá o movimento dos núcleos nesta superfície (note que o termo “superfície” não implica dimensionalidade neste caso). De qualquer forma, o movimento dos núcleos só pode ser calculado depois de determinada a função  $E(\mathbf{R})$ .

O procedimento de encontrar  $E(\mathbf{R})$  pela equação 2.1.2 seria o mais completo; dado  $E(\mathbf{R})$ , poderíamos resolver então a equação 2.1.3. De fato, alguns pacotes computacionais resolvem exatamente este problema (exemplos incluem os pacotes *Dmol*, *Turbomole*, *GAMESS*, *Hondo*, *CADPAC* e *Gaussian*), mas a grandes custos computacionais. Outros métodos, menos custosos matematicamente, aproximam várias das integrais a serem resolvidas no processo de resolução de 2.1.2 por valores empíricos (exemplos deste tipo de abordagem semi-empírica incluem os pacotes *ZINDO*, *MNDO*, *MINDO*, *MOPAC* e *AMPAC*). Os métodos mais convenientes são aqueles que avaliam  $E(\mathbf{R})$  praticamente independentemente de 2.1.2, e apenas com termos empíricos adequados a dadas situações. As funções  $E(\mathbf{R})$  determinadas empiricamente são chamadas comumente de campos de força, ou, em inglês, *forcefields*, e a elas é designada a letra  $V(\mathbf{R})$ , já associada a potenciais, como veremos.

Considerando o método mais simples, o dos campos de força, e apenas o movimento dos núcleos atômicos, a equação de Schrödinger pode ser reduzida à equação padrão de Newton, 2.1.4, dado que os efeitos quânticos terão pouco efeito sobre núcleos, que são objetos relativamente pesados.

$$-\nabla V = m \frac{d^2 \vec{R}}{dt^2} \quad (2.1.4)$$

Deste ponto em diante, associaremos à resolução da equação 2.1.4 o termo Dinâmica Molecular. Um outro tratamento molecular interessante é a Mecânica Molecular, que ignora a evolução do sistema com o tempo e lida com problemas associados à geometria molecular, energias associadas e outras propriedades estáticas do sistema.

## 2.2 – O campo de Força

Para encontrar a superfície de energia potencial necessária para a dinâmica molecular, precisaremos expressar a energia total do sistema em função das coordenadas dos átomos envolvidos. Felizmente, vários pacotes computacionais fornecem maneiras convenientes de definir o campo de força de um sistema. Em especial, a cada átomo de um sistema deve ser atribuído um tipo, dado que diferentes elementos – ou até mesmo os mesmos elementos em diferentes situações – podem se comportar de maneiras diferentes. A cada um destes tipos estão associadas várias formas funcionais de potenciais, como torção, esticamento de ligações químicas, termos de Van der Waals, e assim por diante. Os parâmetros e coeficientes das formas funcionais foram determinados empiricamente e já contêm informações quânticas de maneira experimental, embora de maneira muito limitada e genérica, já que os parâmetros foram encontrados de maneira tal que o campo de força explicasse fenômenos observáveis.

É importante lembrar, neste ponto, que o objetivo do campo de força que nós usaremos é modelar um grande número de situações com e a baixos custos computacionais. Alguns outros tipos de campo de força são mais adequados a um leque menor de situações e têm como objetivo uma grande precisão em sua modelagem e em suas previsões.

De qualquer forma, os métodos associados a campos de força têm suas limitações intrínsecas. A investigação de situações relacionadas a fenômenos como transições eletrônicas, transporte eletrônico e transferência de prótons (reações ácido-base) está além do alcance da maioria dos campos de força.

Apesar destas limitações, os campos de força são muito úteis em sistemas de grande extensão, já que nestes, os métodos quânticos teriam um custo computacional enorme.

Claro que o uso de campos de força não permite a investigação de propriedades quânticas, mas há várias situações nas quais as propriedades de interesse não são quânticas – como no nosso caso. Exemplos destes sistemas incluem macromoléculas e morfologia de cristais.

Além disso, os campos de força também proporcionam um grande nível de controle sobre as energias que regem o sistema. Pode-se decompor o termo  $E(\mathbf{R})$  em potenciais conhecidos de torção, ligações químicas, e assim por diante. É possível aplicar restrições e modificações nestes termos; em particular, é possível restringir alguma espécie de movimento e limitar alguns efeitos para a minimização dos custos computacionais associados à dinâmica molecular. Para um conjunto de átomos, poderíamos restringir os efeitos de torção, por exemplo, devido a alguma peculiaridade do sistema em questão – este tipo de artifício recebe o nome de *restraints*. Uma maneira mais drástica e computacionalmente econômica é restringir o movimento de um conjunto de átomos completamente, método este comumente chamado de *constraints*. Este segundo poderá ser usado no nosso caso, já que o movimento do tubo externo não será de grande importância.


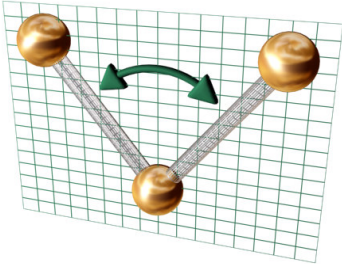
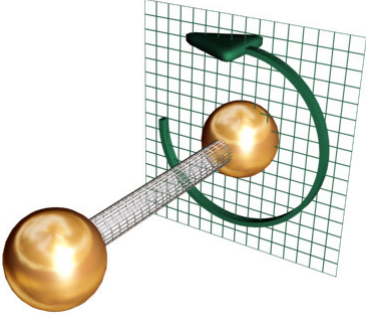
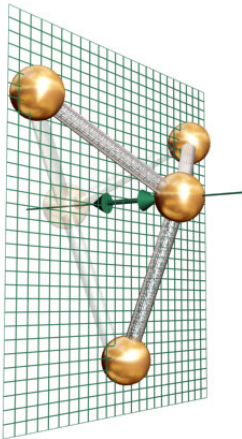
### 2.3 – O Campo de Força expressado matematicamente

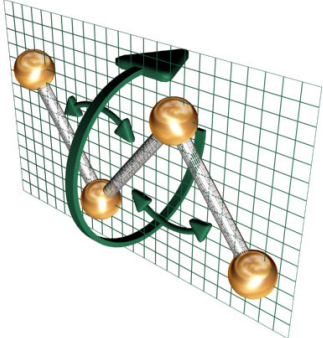
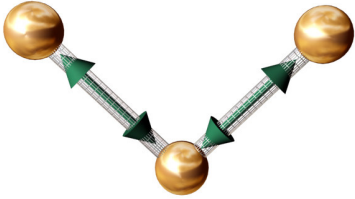
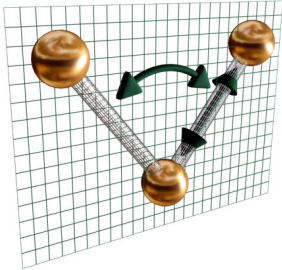
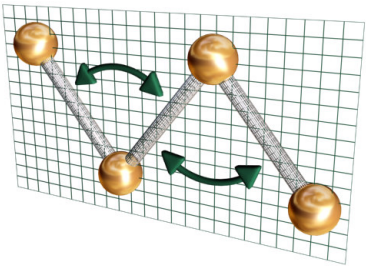
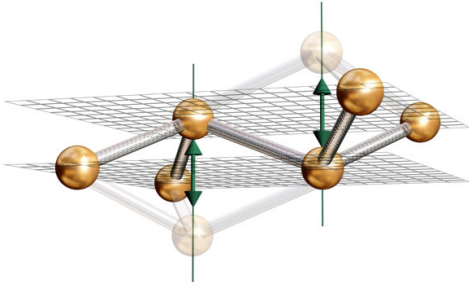
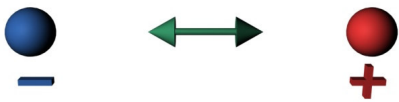
Para encontrar a função  $E(\mathbf{R})$  para um sistema específico, usaremos as informações empíricas de um campo de força e as coordenadas de um dado sistema e encontraremos uma expressão para a energia para uma dada situação, ou seja, uma descrição matemática da superfície de energia potencial vista em 2.1.

A energia potencial para um dado sistema pode ser expressa como a soma dos termos associados às ligações (os termos de valência), os termos cruzados, que representam a interação entre ligações atômicas próximas e cuja implementação é razoavelmente nova, e os termos restantes, que não estão associados às ligações e retratam a interação de átomos quimicamente separados. Atribuindo a estes termos as palavras inglesas *valence*, *crossterm* e *nonbond*, respectivamente, temos a seguinte relação:

$$E_{total} = E_{valence} + E_{crossterm} + E_{nonbond} \quad (2.3.1)$$

A tabela a seguir evidencia algumas considerações possíveis de potenciais.

	<b>Interação</b>	<b>Ilustração</b>	<b>Expressão</b> (forma funcional)
Interações de valência ( $E_{valence}$ )	Esticamento de ligação química		$\sum_b D_b [1 - e^{-(a(b-b_0))}]^2$
	Varição de ângulo entre ligações		$\sum_{\theta} H_{\theta} (\theta - \theta_0)^2$
	Varição de ângulo em uma ligação - torsão		$\sum_{\phi} H_{\phi} [1 + s \cos(n\phi)]$
	Tensão fora-de-plano		$\sum_{\chi} H_{\chi} \chi^2$

Interações cruzadas ( $E_{crossterm}$ )	Torsão ângulo-ângulo		$\sum_{\theta} \sum_{\theta'} F_{\theta\theta'} (\theta - \theta_0) (\theta' - \theta'_0) \cos(\phi)$
	Esticamento quadrático		$\sum_b \sum_{b'} F_{bb'} (b - b_0) (b' - b'_0)$
	Interação ligação-ângulo quadrática		$\sum_b \sum_{\theta} F_{b\theta} (b - b_0) (\theta - \theta_0)$
	Interação ângulo-ângulo quadrática		$\sum_{\theta} \sum_{\theta'} F_{\theta\theta'} (\theta - \theta_0) (\theta' - \theta'_0)$
	Interação tensão fora-de-plano com tensão fora-de-plano		$\sum_{\chi} \sum_{\chi'} F_{\chi\chi'} \chi \chi'$
$E_{nonbond}$	Interação entre cargas (Coulomb)		$\sum_i \sum_{j>i} k \frac{q_i q_j}{r_{ij}}$

Interação entre cargas induzidas (Van der Waals)		$\sum_i \sum_{j>i} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right) \text{ (Lennard-Jones)}$
--	--	---

De acordo com a tabela, então, poderíamos usar a expressão que segue, 2.3.2, para um campo de força exemplificado. Para que 2.3.2 seja de alguma utilidade, é necessário algum conhecimento a respeito dos coeficientes para uma dada situação; na verdade, mesmo a forma matemática funcional de um campo de força sofre modificações, de acordo com o seu uso. Neste sentido, a equação 2.3.2 é apenas um exemplo geral de campo de força. Note que os coeficientes variam não só de acordo com o elemento do átomo, mas de acordo com a sua posição, simetria de ligações químicas, e assim por diante; cada uma das situações deve admitir coeficientes diferentes.

$$\begin{aligned}
V(R) = & \sum_b D_b \left[ 1 - e^{-(a(b-b_0))} \right]^2 + \sum_{\theta} H_{\theta} (\theta - \theta_0)^2 + \sum_{\phi} H_{\phi} [1 + s \cos(n\phi)] + \sum_{\chi} H_{\chi} \chi^2 + \\
& \sum_{\theta} \sum_{\theta'} F_{\theta\theta'} (\theta - \theta_0) (\theta' - \theta'_0) \cos(\phi) + \sum_b \sum_{b'} F_{bb'} (b - b_0) (b' - b'_0) + \sum_b \sum_{\theta} F_{b\theta} (b - b_0) (\theta - \theta_0) + \\
& \sum_{\theta} \sum_{\theta'} F_{\theta\theta'} (\theta - \theta_0) (\theta - \theta'_0) + \sum_{\chi} \sum_{\chi'} F_{\chi\chi'} \chi \chi' + \sum_i \sum_{j>i} k \frac{q_i q_j}{r_{ij}} + \sum_i \sum_{j>i} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right) \quad (2.3.2)
\end{aligned}$$

O campo de força que usaremos para o nosso objetivo de simular fulerenos encapsulados não segue exatamente o exemplo da equação 2.3.2, desenvolvido aqui para dar familiaridade aos conceitos envolvidos. Trata-se de um campo genérico que acompanha o programa de simulações *Cerius2* versão 4.2, e é chamado de *Universal Forcefield*, ou *UFF* no qual os parâmetros são gerados com base em regras fisicamente realistas. Amplos testes indicam que simulações de dinâmica molecular usando o *Universal Forcefield* têm resultados aceitáveis em uma grande gama de situações.

As formas funcionais utilizadas no *UFF* incluem um termo harmônico para o esticamento de ligações. Além deste, este campo descreve as energias associadas a variações de ângulo com três termos de uma expansão de Fourier, e expressa torções e



inversões com termos cosseno de expansões Fourier. As interações de van der Waals são descritas por um típico potencial, o de Lennard-Jones, que tem uma forma  $A r^{(-12)} - B r^{(-6)}$ . As interações eletrostáticas são descritas por monopólos atômicos e um termo coulômbico empiricamente adequado.

## 2.4 – Integração da equação de Newton

Até este ponto, determinamos apenas o campo de força para um dado sistema,  $V(\mathbf{R})$ . Agora resta resolver a equação 2.4.1, versão reescrita aqui de 2.1.4:

$$-\nabla V = m \frac{d^2 \vec{R}}{dt^2} \quad (2.4.1)$$

Vários métodos numéricos integram a equação 2.4.1, mas devemos escolher o mais adequado à nossa situação. Veremos a seguir alguns métodos diferentes. A idéia geral é bastante simples e segue o método das diferenças finitas, no qual as posições e velocidades em um dado instante  $t$  são conhecidas, e devemos calcular as posições e velocidades um momento  $\Delta t$  depois, através de uma fórmula de recorrência, digamos,  $\mathbf{r}(t+\Delta t) = f(\mathbf{r}(t), \mathbf{v}(t), \mathbf{a}(t))$  ou semelhante.

Os critérios que qualquer algoritmo deve satisfazer incluem baixo uso de memória em computadores e pequeno número de avaliações da superfície de energia (estas avaliações requerem grande esforço matemático). Além disso, o algoritmo deve permitir o uso de um intervalo  $\Delta t$  grande, ou o número de iterações necessárias será enorme. Finalmente, o algoritmo também deve ser estável do ponto de vista da conservação de energia.

### Verlet Leapfrog Integrator e Verlet Velocity Integrator

Estes dois métodos e suas variantes são muito amplamente utilizados. Os integradores propostos por Verlet em 1967 têm a vantagem de só precisarem de uma pequena quantidade de memória, apenas avaliar a energia uma vez a cada intervalo de integração e possibilitar o uso de um intervalo de quantização de tempo razoavelmente grande sem problemas. O algoritmo associado ao integrador *Verlet Leapfrog* segue a fórmula de recursão 2.4.3 e utiliza como entrada as grandezas posição, velocidade e aceleração nos instantes  $t$ ,  $t-\Delta t/2$  e  $t$ , respectivamente, como visto em 2.4.2.

$$\text{Entrada de dados: } \vec{r}(t) \quad \vec{v}\left(t - \frac{1}{2}\Delta t\right) \quad \vec{a}(t) \quad (2.4.2)$$

$$\text{Fórmulas de recursão: } \begin{cases} \vec{v}\left(t + \frac{1}{2}\Delta t\right) = \vec{v}\left(t - \frac{1}{2}\Delta t\right) + \Delta t \vec{a}(t) \\ \vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \vec{v}\left(t + \frac{1}{2}\Delta t\right) \\ \vec{a}(t + \Delta t) = \frac{-1}{m} \nabla V|_{\vec{r}(t+\Delta t)} \end{cases} \quad (2.4.3)$$

O principal problema deste algoritmo é evidente nas suas fórmula de recursão: as velocidades e as posições estão sempre fora de sincronia por meio intervalo  $\Delta t$ . Desta forma, só sabemos  $\mathbf{r}(t)$  e  $\mathbf{v}(t+\Delta t/2)$ , e nunca  $\mathbf{r}(t+\Delta t/2)$  ou mesmo  $\mathbf{v}(t)$ . De qualquer forma, é intuitivo que  $\mathbf{v}(t+\Delta t/2)$  seja uma expressão mais adequada para o intervalo  $[t, t+\Delta t]$  do que  $\mathbf{v}(t)$  ou mesmo  $\mathbf{v}(t+\Delta t)$ .

O algoritmo associado ao integrador *Verlet Velocity* não sofre da mesma limitação. Sua fórmulas de recursão podem ser vistas em 2.4.5, e utiliza a posição, velocidade e aceleração em um instante para calcular as mesmas variáveis um certo  $\Delta t$  depois:

$$\text{Entrada de dados: } \vec{r}(t) \quad \vec{v}(t) \quad \vec{a}(t) \quad (2.4.4)$$

$$\text{Fórmulas de recursão: } \begin{cases} \vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \vec{v}(t) + \frac{\Delta t^2 \vec{a}(t)}{2} \\ \vec{a}(t + \Delta t) = \frac{-1}{m} \nabla V|_{\vec{r}(t + \Delta t)} \\ \vec{v}(t + \Delta t) = \vec{v}(t) + \frac{1}{2} \Delta t [\vec{a}(t) + \vec{a}(t + \Delta t)] \end{cases} \quad (2.4.5)$$

De fato, os dois métodos de Verlet são os mais úteis. Outros métodos ainda existem, incluindo o chamado ABM4, de “*Adams-Bashforth-Moulton*” de quarta ordem, ou o bastante robusto mas computacionalmente exigente Runge-Kutta de 4ª ordem. Este último é um dos métodos mais antigos de resolução de equações diferenciais.

Aqui, vamos fazer uso exclusivamente do Integrador Verlet Leapfrog, um método rápido e eficiente para os nossos propósitos. As simulações de dinâmica molecular do programa utilizado, o *Cerius2*, já usam como padrão o método de Verlet dessincronizado. O usuário deve apenas especificar o intervalo  $\Delta t$  a ser considerado.

Para escolher o intervalo  $\Delta t$ , devemos ter em mente que um intervalo pequeno implicaria um número demasiado grande de iterações, e um intervalo grande causaria grandes imprecisões. Para evitar estas últimas, certamente deve-se considerar um intervalo menor do que o movimento de maior frequência em um dado sistema. Na enorme maioria dos modelos orgânicos, a maior frequência vibracional é a da oscilação da ligação C-H, cujo período é da ordem de  $10^{-14}$ s (10fs). Note que os algoritmos de integração por diferenças finitas geralmente assumem que  $\mathbf{r}$  e  $\mathbf{v}$  são aproximadamente constantes durante o intervalo  $\Delta t$ ; para que isto seja uma suposição razoável, é recomendável dividir um período vibracional em 8 a 10 partes. Desta forma, um  $\Delta t$  aceitável para a maioria dos modelos

orgânicos estaria na faixa de 0,5 a 1,0 fs. Existem vários erros de integração que podem aparecer para  $\Delta t$ 's exageradamente grandes, mas não entraremos em mais detalhes aqui. Em geral, se possível computacionalmente, é sempre melhor considerar  $\Delta t$  bastante pequeno.

## 2.5 – Os Ensembles Estatísticos

A resolução da equação de Newton, 2.4.1, conforme a descrição da seção 2.4, permite a exploração da superfície de energia constante do sistema. Porém, a maioria dos sistemas reais não está isolada de tal maneira do resto do universo. Alguns processos decorrem de maneira adiabática; outros, de maneira isotérmica ou isobárica, dependendo do material analisado. As diferentes considerações de grandezas constantes recebem o nome de *Ensembles Estatísticos*. Aqui, veremos dois dos mais interessantes para o nosso caso, o microcanônico e o canônico.

O ensemble microcanônico considera a energia total do sistema uma constante, assim como o volume do sistema. A resolução direta da equação 2.4.1 sem mais correções de variáveis já está de acordo com o ensemble microcanônico. Porém, o uso do Integrador Verlet Leapfrog causa algumas flutuações no sistema devido à sua falta de sincronia – note que a energia cinética e a energia potencial estão também fora de fase e dessincronizadas. Para manter as flutuações dentro de níveis aceitáveis, é interessante considerar um nível de controle sobre a temperatura. Tendo isso em mente, o programa *Cerius2* possibilita ao usuário a entrada de um intervalo no qual a temperatura deve ser mantida através de multiplicadores aplicados periodicamente às velocidades do sistema. Veremos mais alguns detalhes sobre as medidas e o ajuste da temperatura termodinâmica (não cinética) em um sistema de escala molecular. De qualquer forma, o ensemble microcanônico é o mais adequado à investigação da superfície de energia constante quando não se deseja a perturbação causada pelos outros ensembles.

O ensemble canônico considera a temperatura e o volume do sistema como sendo constantes. As medidas e o ajuste da temperatura são ainda mais importantes para o ensemble canônico do que para o microcanônico, obviamente. O controle direto da

temperatura durante simulações de dinâmica molecular que seguem o ensemble canônico também acontece periodicamente através da aplicação de multiplicadores à velocidade, mas, em geral, um outro método de controle, mais sutil, é empregado continuamente durante a fase da simulação. O ensemble canônico é o mais adequado em sistemas nos quais se deseja saber a conformação de estruturas no vácuo.

Outros ensembles, que consideram pressão constante, estresse estrutural constante ou entalpia constante, também existem, mas as suas aplicações em geral limitam-se a sistemas periódicos (i.e., estruturas cristalinas). Evidentemente, para a dinâmica molecular em tais sistemas, seria necessário desenvolver uma metodologia para a medida e o controle das variáveis pressão, entalpia e estresse estrutural. Aqui, vamos nos limitar à metodologia de medição e controle da temperatura termodinâmica apenas.

#### A Temperatura de um sistema: introdução

A temperatura é uma variável de estado tipicamente macroscópica cuja relação com o microscópico se dá através da energia cinética das partículas de um sistema. De fato, a temperatura e a distribuição de velocidades atômicas são relacionadas através da conhecida fórmula de Maxwell-Boltzmann, 2.5.1. (note que  $k$  é a constante de Boltzmann e  $m$  é a massa atômica)

$$f(v)dv = \left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} e^{-\left(\frac{mv^2}{2kT}\right)} 4\pi v^2 dv \quad (2.5.1)$$

Para iniciar uma simulação de dinâmica molecular, precisamos das condições iniciais para a equação 2.4.1, ou seja, posições e velocidades iniciais. Geralmente, é conveniente determinar apenas as posições, e distribuir as velocidades aleatoriamente de acordo com 2.5.1 e uma dada temperatura inicial. Na verdade, enquanto a distribuição dos módulos das velocidades é adequadamente descrita por 2.5.1, é mais apropriado utilizar

uma distribuição gaussiana, como em 2.5.2, para distribuir as velocidades nas diferentes coordenadas cartesianas (x,y,z).

$$g(v_x)dv_x = \left(\frac{m}{2\pi kT}\right)^{\frac{1}{2}} e^{\left(-\frac{mv_x^2}{2kT}\right)} dv_x \quad (2.5.2)$$

### Medindo a Temperatura de um sistema

A relação entre a temperatura e a energia cinética das partículas de um sistema pode ser descrita através do princípio de equi-partição da energia, que nos diz que a cada grau de liberdade pode ser associada uma energia média de  $kT/2$ . Escrevendo esta relação, chegamos na equação (2.5.3), escrita para cada uma das partículas, de  $i=0,\dots,N$ , e considerando o momento associado a cada uma delas como sendo  $p_i$ . Nesta equação,  $N_f$  representa o número de graus de liberdade do sistema considerado.

$$\left\langle \sum_i^N \frac{p_i^2}{2m} \right\rangle = \langle K \rangle = \frac{N_f kT}{2} \quad (2.5.3)$$

Para continuar esta análise, é conveniente definirmos uma temperatura cinética instantânea segundo a equação 2.5.4. Considerando então a equação 2.5.3 novamente, temos a relação imediata de que a média da temperatura instantânea é automaticamente a temperatura termodinâmica usual do sistema.

$$T_{ins} = \frac{2K}{N_f k} \quad (2.5.4)$$

Com as informações apresentadas, podemos calcular a temperatura de um dado sistema de N partículas com a expressão mais condensada, 2.5.5.

$$\frac{(3N-6)kT}{2} = \sum_{i=1}^N \frac{m_i v_i^2}{2} \quad (2.5.5)$$

Note que consideramos 3N-6 graus de liberdade, pois ignoramos a translação e a rotação ao redor do centro de massa do sistema.

### O Controle da Temperatura

Vamos apresentar dois métodos de controle de temperatura. O primeiro, que chamaremos de Multiplicador de Temperaturas, é bastante drástico e simplesmente multiplica todas as velocidades do sistema por um fator de ajuste. Obviamente, a temperatura pode ser controlada de forma bastante rígida, mas este método suprime as flutuações naturais do sistema e não é termodinamicamente realista. O multiplicador das velocidades utilizado pelo programa *Cerius2* é dado pela equação 2.5.6. Nesta equação,  $T_0$  representa a temperatura desejada,  $T_{inst}$  representa a temperatura instantânea de 2.5.4, e  $\langle T \rangle$  representa a média da temperatura.

$$\lambda = \begin{cases} \sqrt{2\left(\frac{T_0 - \langle T \rangle}{T_{inst}}\right) + 1}, & \text{se } \left[2\left(\frac{T_0 - \langle T \rangle}{T_{inst}}\right) + 1\right] \geq 0 \\ 1, & \text{caso contrário} \end{cases} \quad (2.5.6)$$

Evidentemente, o uso da equação 2.5.6 afeta a energia total do sistema de maneira imediata. Para o ensemble canônico, no qual a temperatura é constante, o uso deste multiplicador é inadequado. Uma abordagem mais realista é a do método de Berendsen para o acoplamento do sistema com um reservatório térmico, de acordo com o qual o multiplicador  $\lambda$  pode ser expresso como na equação 2.5.7. Nesta equação,  $\Delta t$  representa a diferença finita de tempo para a integração numérica,  $\tau$  é um tempo de relaxamento característico,  $T$  é a temperatura instantânea (2.5.4) e  $T_0$  é a temperatura desejada.

$$\lambda = \sqrt{1 - \frac{\Delta t}{\tau} \left( \frac{T - T_0}{T} \right)} \quad (2.5.7)$$

Em boa aproximação, o uso da equação 2.5.7. representa a utilização de um ensemble canônico, no qual a temperatura é mantida constante em  $T_0$ .

## 2.6 – Minimizadores de Energia

Além da dinâmica molecular, que representa o comportamento de um sistema dada uma situação inicial, temos também um outro grande recurso disponível: os minimizadores de energia. Tais ferramentas são capazes de encontrar a conformação de energia mínima de um sistema.

Evidentemente, a primeira alternativa que se sugere para tal tarefa é um método semelhante à dinâmica molecular, com as velocidades zeradas a cada passo. Desta forma, acompanhamos o negativo do gradiente da função superfície de energia e chegamos ao mínimo, onde o gradiente é nulo. Este método não é possível exatamente desta maneira, pois iria necessitar de muitas avaliações desse gradiente.

Desta forma, a primeira alternativa razoável que encontramos, a chamada procura por descidas mais íngremes (*steepest descent method*), é a seguinte: consideremos um dado



campo escalar  $F$  a ser minimizada (no caso, a própria superfície de energia) e um ponto inicial  $(x_0, y_0)$ . Um método consiste em minimizar a função paramétrica:

$$f : R \rightarrow R$$

$$f(t) = (x_0, y_0) + t \nabla \vec{F}(x_0, y_0) \quad (2.6.1)$$

Essencialmente, desta forma, minimizaremos a função  $F$  ao longo de uma reta determinada pelo seu gradiente inicial, e colocaremos no mínimo o ponto inicial da próxima iteração, avaliada da mesma maneira.

Matematicamente, é simples ver que todas as direções subseqüentes estarão perpendiculares às anteriores, dado que o componente do gradiente avaliado no ponto mínimo ao longo da reta de minimização será nulo na direção

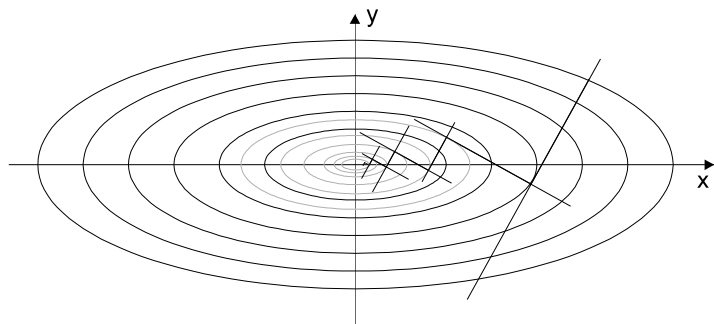


Figura 2.6.A: O método das descidas íngremes, ou mais inclinadas (*steepest descents*).

da reta e o gradiente da função aponta perpendicularmente, portanto, à reta. Desta forma, a procura por linhas extrai toda a energia de uma direção a cada iteração, o que torna o método bastante interessante.

Existem algumas desvantagens associadas a este método. A minimização da função 2.6.1 requer de 3 a 10 avaliações da superfície de energia, e, portanto, o método não é computacionalmente barato. Este é um caso particular da mais genérica procura por linhas, onde o primeiro vetor direção da primeira reta não é definido.

Para melhor a eficiência deste método, é possível utilizar um algoritmo até mais simples do que este: considerando a reta ao longo do gradiente inicial, tentamos um ponto qualquer na direção de  $-\nabla F$  e avaliamos o valor de  $F$  e, se este for menor do que o inicial, consideramos o ponto escolhido como o novo ponto inicial para outra iteração. Desta forma, é possível reduzir o número de avaliações de  $F$  drasticamente. Este método é chamado de minimização por descidas mais íngremes sem procura por linhas (do inglês

bastante apropriado “*minimization following a steepest-descent path without line searches*”).

De qualquer forma, as vantagens e desvantagens dos métodos de descidas mais íngremes vêm da sua dependência do gradiente da função a ser minimizada. O gradiente próximo do mínimo é zero, o que pode reduzir a velocidade de convergência quando na sua proximidade. Porém, o método é bastante eficiente em encontrar conformações de menor energia, mesmo para sistemas irregulares e nada harmônicos. Estes dois argumentos acabam tornando as procuras por descidas mais íngremes os métodos mais utilizados durante os primeiros estágios de uma minimização, quando o sistema está longe de uma configuração harmônica; depois da fase inicial de minimização, é conveniente o uso de outro algoritmo de minimização.

Alguns algoritmos sofisticados são variantes dos baseados em procura por linhas. Especificamente, estes algoritmos procuram aumentar a eficiência dos métodos vistos anteriormente considerando correções na escolha da nova direção de minimização. Tais correções são interessantes devido à natureza oscilante da trajetória usual do método das descidas mais íngremes. Os métodos que fazem uso destas correções são chamados de Gradientes Conjugados, e em geral explicitam o novo vetor direcional para as iterações seguintes como sendo combinação linear do gradiente atual e do anterior, de acordo com a equação 2.6.2, que nos dá o vetor direção novo,  $\mathbf{h}$ : ( $\mathbf{F}$  é a função a minimizar)

$$\vec{h}_{n+1} = \nabla \vec{F}(x_{n+1}, y_{n+1}) + \gamma_n \nabla \vec{F}(x_n, y_n) \quad (2.6.2)$$

A escolha do fator de combinação  $\gamma$  é bastante relevante. Veremos aqui duas possibilidades. A primeira é a da equação 2.6.3, e é chamada de método de Polak-Ribiere. A segunda é a da equação 2.6.4, o chamado método de Fletcher-Reeves. Note que todos os produtos são, na realidade, produtos escalares.

$$\gamma_n = \frac{(\nabla \bar{F}(x_{n+1}, y_{n+1}) - \nabla \bar{F}(x_n, y_n)) \nabla \bar{F}(x_{n+1}, y_{n+1})}{(\nabla \bar{F}(x_n, y_n))^2} \quad (2.6.3)$$

$$\gamma_n = \frac{(\nabla \bar{F}(x_{n+1}, y_{n+1}))^2}{(\nabla \bar{F}(x_n, y_n))^2} \quad (2.6.4)$$

A direção dada por 2.6.2 é então utilizada de acordo com o método das descidas mais íngremes. Estes dois métodos têm a propriedade de gerar, além de gradientes mutuamente perpendiculares, como era de se esperar, vetores direcionais que são conjugados a todos os vetores direcionais anteriores. Genericamente, os métodos de gradiente conjugado convergem em N passos, onde N é o número de graus de liberdade do sistema.

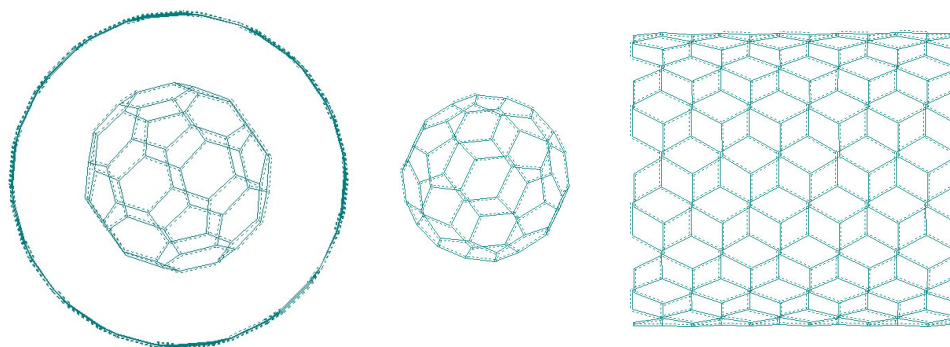
Outros métodos mais custosos computacionalmente (e mais precisos) também existem. Especificamente, os métodos de Newton-Raphson são bastante pesados, e fazem uso da segunda derivada da função a ser minimizada. Devido ao nosso uso limitado de minimizadores de grande precisão, não explicaremos estes métodos aqui.

## 2.7 – O Encapsulamento de Fullerenos

Recentemente, Hornbaker et. al. [1] mediram as propriedades eletrônicas de estruturas chamadas “peapods”, que nada mais eram do que moléculas de fullerenos – especificamente, o C60 – encapsuladas dentro de nanotubos de tamanhos apropriados. Estas estruturas são de interesse na eletrônica em diversos aspectos, e os “peapods” se formam espontaneamente quando tubos de diâmetro apropriado são colocados em contato com muitas moléculas de C60. As propriedades mecânicas do sistema e o seu processo de encapsulamento serão o nosso objeto de análise inicial, para uma melhor compreensão dos métodos de dinâmica molecular. Trata-se de um estudo para treinamento.

É importante percebermos que, até este ponto, nada mais foi feito do que uma introdução aos métodos utilizados para resolver um problema, o do encapsulamento e do comportamento mecânico de moléculas de C60 encapsuladas em nanotubos de diâmetros apropriados. Inicialmente, pretendemos investigar em quais tubos o C60 é encapsulado com mais facilidade, de acordo com o diâmetro dos tubos. Evidentemente, existe uma miríade de opções de tubos, especialmente se considerarmos os tubos quirais, nos quais  $n \neq m$  e  $m \neq 0$ .

Como critério para a escolha, devemos pensar que uma molécula de C60 tem, tipicamente, um diâmetro da ordem de  $6,9\text{\AA}$ . Devemos ainda considerar uma certa diferença nos diâmetros entre o tubo “encapsulante” e o C60, de tal modo que o C60 tenha alguma mobilidade dentro do tubo. Tendo isto em mente, esperamos que uma faixa de 11 a  $13\text{\AA}$  seja aceitável – de qualquer forma, se não estiver, podemos fazer mais testes. Consultando os nossos geradores de nanotubo, os tubos não-quirais mais adequados pareceriam ser os tubos (9,9) e (15,0). A figura 2.7.A mostra a configuração inicial da nossa simulação de dinâmica molecular para o tubo (9,9) e a figura 2.7.B mostra o mesmo para o tubo (15,0). Da forma como estão dispostas as moléculas, assumimos que as forças de Van der Waals irão provocar uma força atrativa que causará o encapsulamento automaticamente. Sabemos que, na prática, o encapsulamento acontece sem grandes esforços; logo, assumimos que seremos capazes de descrever o processo com as técnicas descritas aqui. Teremos a vantagem adicional de um nível grande de controle do sistema; poderemos prever qual tubo será mais adequado mais tarde, depois de uma comparação de vários casos.



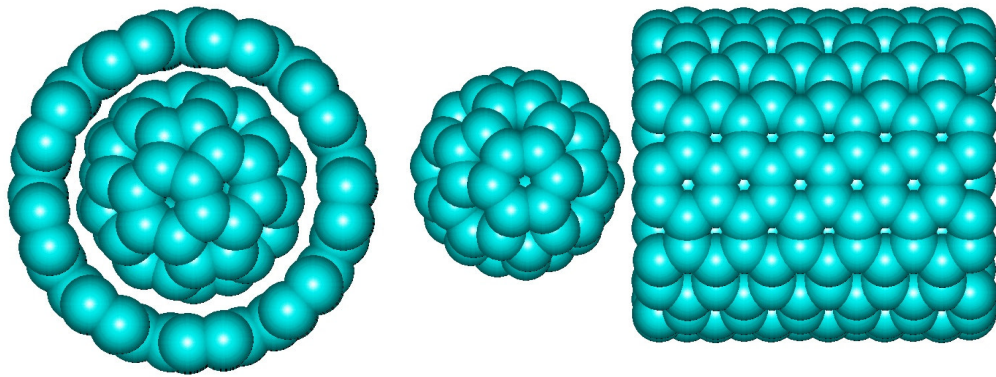


Figura 2.7.A: Configuração inicial para a simulação de dinâmica molecular de encapsulamento do C60 em um tubo Armchair (9,9).

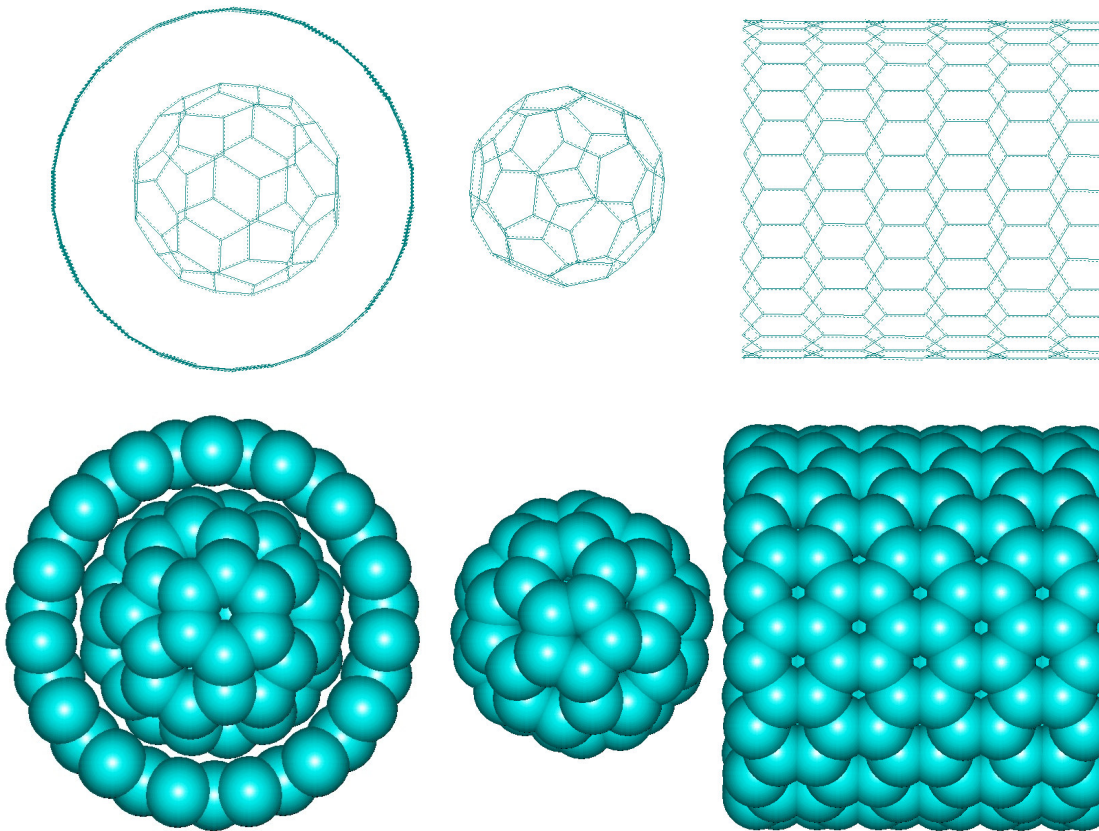


Figura 2.7.B: Configuração inicial para a simulação do encapsulamento do C60 em um tubo de (15,0).

A inspeção da figura 2.7.B revela que já parece improvável que aconteça algum encapsulamento para esta configuração; o tubo parece muito pequeno. Decidimos tentar ainda assim simular esta situação para analisar o comportamento do fulereno nestas posições e condições. Note que a figura 2.7.A parece indicar que o encapsulamento no tubo (9,9) é mais provável, pois parece haver mais espaço.

Gerando então os tubos relevantes e utilizando uma simulação com baixo controle de temperatura (mantendo a temperatura dentro de níveis de tolerância), e de acordo com o ensemble microcanônico, com um integrador Verlet Leapfrog, constatamos que o encapsulamento proposto na figura 2.7.B de fato não ocorre. É possível que o fulereno mova-se inicialmente em direção ao tubo, mas, à medida em que a proximidade das paredes do nanotubo aumenta, parece existir uma situação de equilíbrio na qual o fulereno tende a ficar, como que tampando o tubo de certa forma.

Além disso, o encapsulamento com o tubo (9,9) também não ocorre, sendo que, para esta situação, o C60 também parece encontrar uma conformação de energia mínima logo na entrada do tubo, permanecendo lá. Os resultados parecem indicar que tubos de diâmetro maior sejam adequados.

Pensando desta forma, tentamos usar os tubos (16,0) e (10,10) para o encapsulamento. O tubo (16,0) ainda é muito pequeno e precisamos recorrer ao tubo (17,0); porém, no tubo (10,10), usando o mesmo método, fomos capazes de constatar a entrada

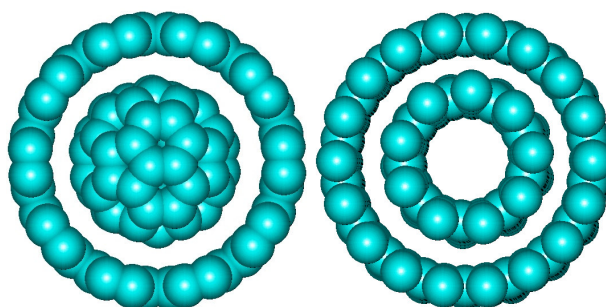


Figura 2.7.C.: Comparação dos dois encaixes: um oscilador nanotubo-nanotubo com tubos (9,0) e (18,0) (direita) e C60 encapsulado em um tubo (10,10) (esquerda).

do C60 para dentro do tubo, e conseqüente encapsulamento devido a efeitos de capilaridade; de fato, uma vez contido dentro do tubo, o C60 passa a realizar um movimento inicial oscilatório dentro do nanotubo, pelo menos nos instantes iniciais (veremos logo que a oscilação não é estável). A figura 2.7.D revela o que acontece no caso do tubo Armchair (10,10). Note que configurações iniciais nas quais o C60 está a razoáveis distâncias da entrada do nanotubo também não parecem ter efeito algum, dado que a

interação Van der Waals é bastante fraca e só é capaz de exercer o seu efeito para fulerenos próximos da entrada do tubo. Isto indica que o fullereno deve primeiramente se aproximar da entrada do tubo, possivelmente por agitações térmicas, e então ser encapsulado. O encaixe da molécula de C<sub>60</sub> pode ser visto na figura 2.7.C, na qual também se vê a mesma perspectiva de um oscilador nanotubo - nanotubo conhecido de encaixe perfeitamente deslizante. Os encaixes parecem semelhantes; note, é claro, que o sistema nanotubo-nanotubo apresenta comportamento oscilatório, enquanto o sistema nanotubo-C<sub>60</sub> não.

Ainda resta a pergunta: como acontece exatamente o processo de encapsulamento múltiplo? Tentando responder a esta questão, colocamos dois fulerenos em duas extremidades de um tubo (10,10). Conforme já era de se esperar, ambos se moveram em direção ao centro do tubo, e sofreram repulsão mútua devido à aproximação; além disso, nenhum dos dois saiu do tubo e ambos permanecem a uma distância centro-a-centro de cerca de 9,8Å, com uma pequena flutuação; a conformação final é a da figura 2.7.E e foi obtida considerando dois fulerenos independentes, um em cada extremidade do tubo, e encontrando-se a conformação de mínima energia. Alguns testes também foram feitos para sistemas com mais de dois fulerenos; nestes, posicionamos os fulerenos individualmente nas proximidades da abertura do tubo e encontramos conformações de mínima energia para um número crescente de C<sub>60</sub>s. De fato, usando esta abordagem, constatamos que a primeira molécula inserida sempre assumia uma posição conveniente, próxima da entrada do tubo por uma distância equivalente a um valor entre dois e três cilindros unitários de altura. Inserindo mais fulerenos, o fullereno inicial se deslocava para uma posição mais interior e mais distante da entrada, e o último fullereno inserido ocupava praticamente a mesma posição que o primeiro C<sub>60</sub> inserido ocupou, a aproximadamente dois cilindros unitários e meio de distância da entrada do tubo. Assim, o tubo inteiro é preenchido, dado que esta é uma configuração energeticamente favorável ao sistema inicial, no qual o tubo estava vazio.

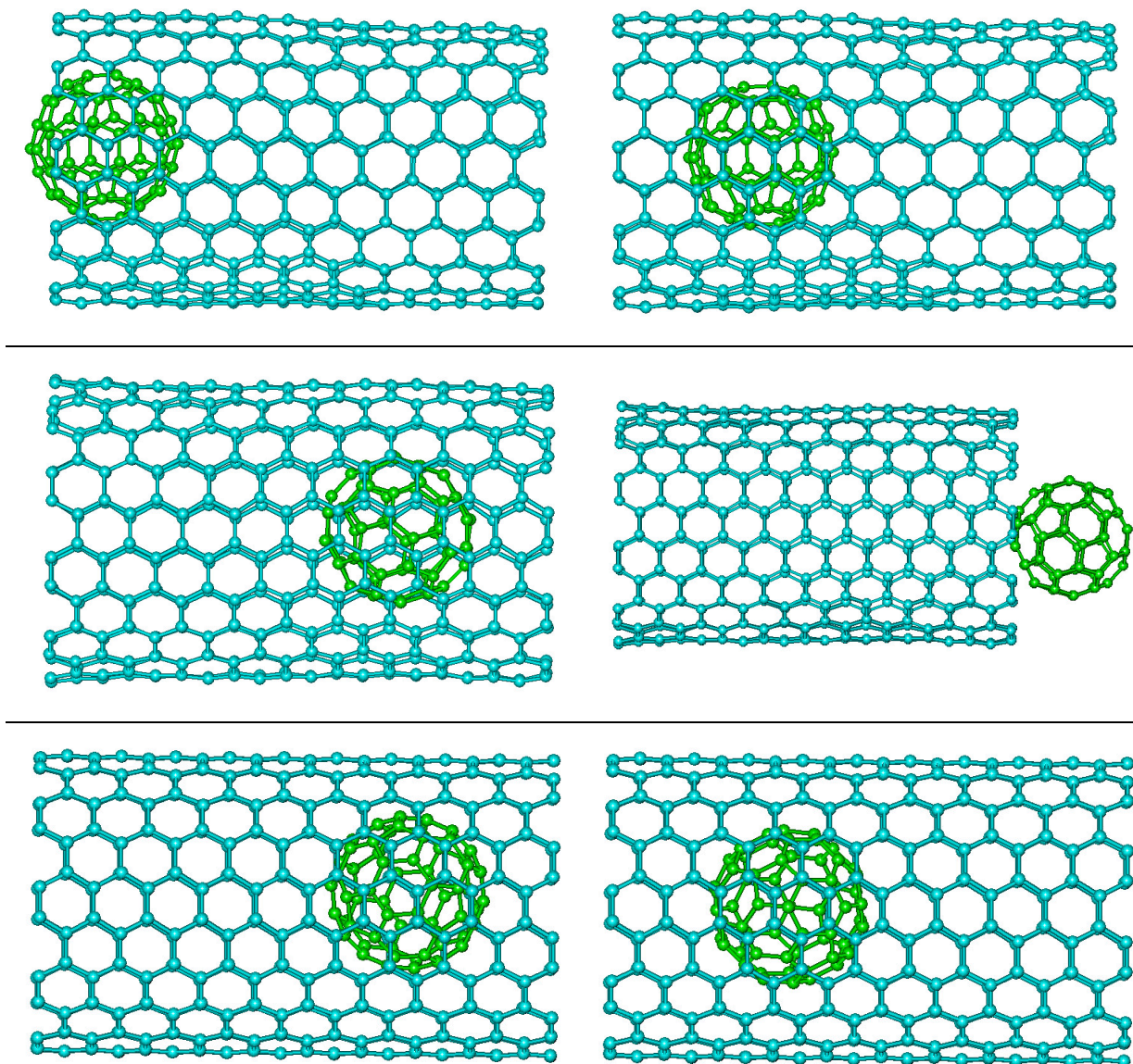


Figura 2.7.D: (dinâmica molecular) - O Deslizamento de uma molécula de C60 através de um tubo armchair (10,10). A força que atua durante o deslizamento na parte mais interna no tubo é extremamente pequena, mas nas extremidades, é suficiente para fazer o fulereno voltar.



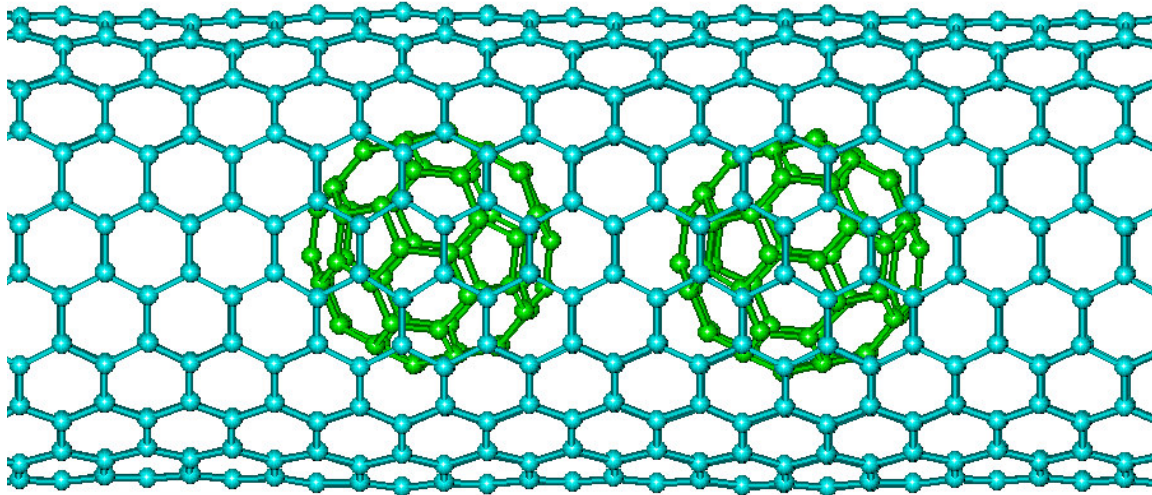


Figura 2.7.E: A conformação espacial mais adequada para dois fulerenos.

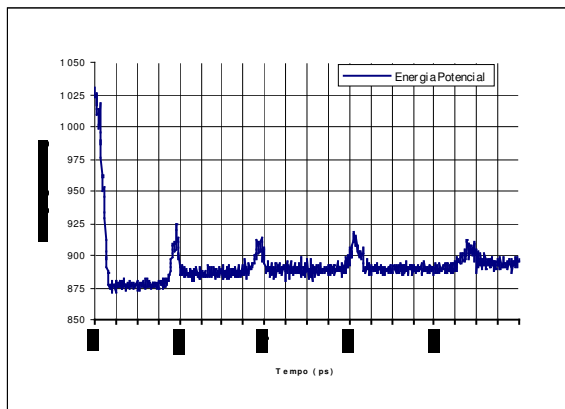


Figura 2.7.F: Energia potencial de acordo com o tempo.

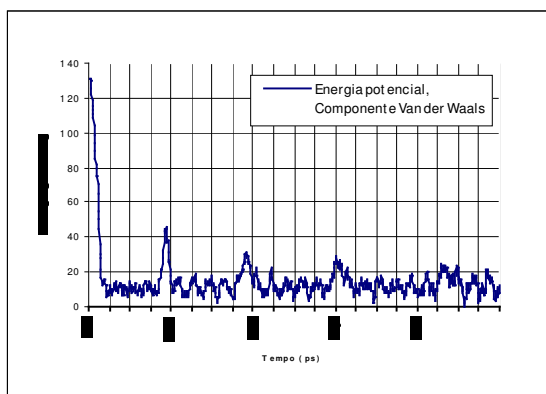


Figura 2.7.G: Componente Van der Waals da Energia total, de acordo com o tempo.

Com as simulações, também fomos capazes de criar gráficos de grande utilidade, como o visto na figura 2.7.F, onde podemos ver a energia potencial do sistema de acordo com o tempo, e o da figura 2.7.G, onde vemos a componente de energia de Van der Waals do sistema. Note como a força de Van der Waals é um dos principais fatores relevantes ao sistema, já que os picos na energia potencial são consequência direta da componente Van der Waals, vista em 2.7.G, e estes picos ocorrem simultaneamente. Além disso, embora não tenhamos mencionado isto antes, o período de oscilação inicial é de aproximadamente 12ps, e cresce de acordo com o tempo, o que fortemente sugere que, apesar do tubo não ser maior do que o necessário, a oscilação não pode ser

sustentada por muito tempo. Isto equivale a uma frequência inicial de oscilação de cerca de 8Ghz. Note que, para o sistema oscilatório tubo-tubo, o nosso próximo objeto de análise depois desta etapa de treinamento, as frequências são consideravelmente maiores, já que as forças de Van der Waals para um sistema tubo-tubo também são mais intensas.

Uma outra consequência interessante dos gráficos vistos é que a energia potencial parece ser aproximadamente constante ao longo do interior do tubo. Desta forma, o C60 executará um movimento uniforme dentro do tubo, se não houver mais impedimentos. Note que a instabilidade do sistema sugere troca de momento com o nanotubo, o que lentamente reduz a energia cinética do sistema e amortece a oscilação.

## 2.7. Análise dos Resultados e Conclusão

A nossa investigação de encapsulamento transcorreu sem grandes surpresas. Constatamos que o encapsulamento do C60 não parece ser possível em tubos (9,9), (15,0) e (16,0) e abaixo, dos tipos Zigzag e Armchair. O encapsulamento acontece automaticamente, porém, para tubos (10,10) e (17,0) e acima; porém, constatamos apenas um movimento oscilatório temporário. Na verdade, a interação é tão fraca que o fulereno tende a se mover em movimento uniforme quando no interior do tubo. Quando se considera vários fulerenos, constatamos a formação das estruturas procuradas. Neste caso, os fulerenos parecem encontrar uma conformação de menor energia entre si e dentro do nanotubo, o que resulta nas estruturas “peapod” (do inglês “*vagem de ervilhas*”)

Comparando os nossos resultados com os dados obtidos por Hornbaker et. Al., vemos que as nossas experiências virtuais parecem ser compatíveis com as observações de Hornbaker et. Al. O grupo relatou uma distância entre os centros dos vários C60 encapsulados em um nanotubo como estando por volta de 10,5Å, com um desvio padrão de 1,0Å; a distância que encontramos, cerca de 9,8Å, para a conformação de dois fulerenos encapsulados é compatível com este dado.

O próximo passo na nossa pesquisa atual é a investigação da conformação de fulerenos dentro de tubos de diâmetro maior do que o mínimo. Especificamente, esperamos

encontrar padrões tridimensionais de conformação para tubos com diâmetro de 1,5 até 2,5 vezes o diâmetro do C60. Como exemplo, poderíamos esperar um padrão zig-zag planar para tubos não-quirais com uma razão de diâmetros pequena, digamos, de 1,1 até 1,5 – mas precisaremos confirmar tais hipóteses com simulações. Tubos quirais devem apresentar comportamento diferente, já que existe o fator alinhamento entre faces do fulereno e faces do tubo – consequência das interações de Van der Waals – que deve provocar um padrão possivelmente helicoidal para a conformação dos fulerenos em tubos quirais de diâmetro maior. Enfim, vamos continuar a investigação da conformação do C60 em nanotubos. Além disso, pretendemos também reproduzir o resultado já conhecido a respeito do comportamento de nanoosciladores baseados em nanotubos de carbono, o que deve ser bastante simples, dado o conhecimento adquirido nesta fase de treinamento. Sabe-se que os nanotubos devidamente encaixados constituem um oscilador de grande estabilidade; pretendemos testar e confirmar este resultado.

## *Referências*

- [1] J. Cummings and A. Zettl, *Science* **289**, 602 (2000)
- [2] S. Berber , Young-Kyun Kwon and David Tomanék, *Phys. Rev. Lett.* **88**, 185502-1 (2002)
- [3] Hornbaker et al, *Science* **267**, 838 (2001)
- [4] UFF\_Valbond1.1 Molecular Force Field, disponibilizado pela Accelrys, Inc. como parte dos pacotes computacionais Materials Studio e Cerius2.  
<http://www.accelrys.com>.

## *Bibliografias*

1. Manual – **Cerius 2**, Versão 4.0, Abril 1999