

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

MÁRCIO JOSÉ MELLO DA SILVA

**Integração Disambiguation Canvas e
Unitymol: Interação 3D em ambientes
densos**

Monografia apresentada como requisito
parcial para a obtenção do grau de Bacharel
em Ciência da Computação

Orientador: Prof. Dra. Luciana Porcher
Nedel

Co-orientador: Prof. Dr. Anderson Maciel

Porto Alegre
2014

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Graduação: Prof. Sérgio Roberto Kieling Franco

Diretor do Instituto de Informática: Prof. Luis da Cunha Lamb

Coordenador do Curso de Ciência de Computação: Prof. Raul Fernando Weber

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

“A Verdadeira batalha só começa quando você acha que não dá mais.”
— MESTRA GENKAI (YU YU HAKUSHO)

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer ao meu Pai que esforçou-se por anos para me trazer até aqui. Gostaria de agradecer também a minha família por todo o apoio, compreensão e paciência com a minha rotina de trabalho e estudo nesses últimos anos.

Agradeço também por toda ajuda e colaboração prestada pelos colegas do grupo de computação gráfica, processamento de imagens, interação e visualização da UFRGS. Dentre muitos fatores, estes auxiliaram-me com problemas de implementação quando meu conhecimento não era suficiente, com ideias novas e soluções mais eficientes para os meus problemas, sou grato também por todo o coleguismo e amizade que ofereceram e, dessa maneira, tornaram essa tarefa um pouco menos árdua. Agradeço em especial os colegas, Theodoro Mota, Victor Adriel, Dennis Giovani, Bruno Pagno, Gerson Groth, Augusto Luengo, Juliano Franz e Roger Leite, pois a ajuda deles foi crucial para a realização deste trabalho.

Agradeço também aos meus Orientadores, Luciana Nedel e Anderson Maciel, que nos últimos dois anos estiveram sempre a disposição para ajudar, mesmo quando o tempo não os favoreciam sempre encontravam uma maneira para me auxiliar. Sou grato não só pelos ensinamentos que vão desde como iniciar um projeto de pesquisa até a escrita precisa de um artigo como requer a ciência, mas também por toda a amizade nas reuniões do grupo, na convivência do dia a dia e nas viagens para conferências que com certeza colaboram muito no meu aprendizado.

Finalmente, agradeço aos financiadores de projetos e bolsas que prestaram-me auxílio financeiro nesses dois anos e também patrocinaram as viagens para conferências, meu sinceros agradecimentos ao CNPQ, FAURGS e Microsoft.

RESUMO

Os sistemas virtuais: simuladores, jogos, *serious games*, estão cada vez mais reais e são capazes de renderizar imagens em alta qualidade, cada vez mais realísticas. Entretanto, ainda persiste o problema de como interagir nesses ambientes virtuais de maneira natural e livre de erros. Interagir pode ser navegar nesses ambientes, manipular e selecionar objetos. A seleção apresenta três grandes problemas: precisão, ambiguidade e complexidade. Em ambientes densos - com muitos objetos - esses problemas intensificam-se ainda mais, pois apresenta um elevado grau de oclusão, devido ao grande número de objetos presentes no ambiente.

Neste trabalho propusemos a integração da ferramenta de visualização molecular Unitymol com a técnica de interação Disambiguation Canvas. Unitymol é um editor desenvolvido na Unity3D que permite a visualização e edição de moléculas. O Disambiguation Canvas é uma técnica de seleção por apontamento com um smartphone, baseada em dois passos, rápida e precisa. A integração dos dois sistemas visa uma interação mais rápida e intuitiva no Unitymol, pois além de incluir seleção de átomos no editor, a molécula que tradicionalmente é manipulada com o mouse, poderá ser movida com o smartphone através de movimentos mais intuitivos. Além disso, com essa integração propusemos uma alternativa manual para o *algoritmo de busca* – primeira etapa da maioria dos algoritmos de docagem molecular.

Palavras-chave: Interação. Disambiguation Canvas. Unitymol. Seleção.

ABSTRACT

Virtual Systems: simulators, games, serious games, are even more real, and they are able to render high quality realistic images, but still remains the problem of how to interact in these virtual environments naturally and with no errors. Interaction can be navigate in such environments, manipulate and select objects. Selection presents three majors problems accuracy, ambiguity and complexity. In cluttered environments these problems became even more stronger because it has more occlusion due the larger number of objects present in this environment.

In this work we propose the integration of the molecular visualization tool Unitymol with the interaction technique Disambiguation Canvas. Unitymol is a tool developed in Unity 3D that allows molecular visualization and edition. Disambiguation Canvas is a selection pointing technique by a smartphone, based on two steps, fast and precise. The integration of the two systems aims for a faster and more intuitive interaction in Unitymol, besides, selection inclusion in the molecular viewer, now the molecule that traditionally is manipulated with the mouse, now can be manipulate with the smartphone with more intuitive movements . Moreover, with this integration we propose a manual alternative for the search algorithm – the first stage of most molecular docking algorithms.

Keywords: Interaction, Disambiguation Canvas, Unitymol, Selection.

LISTA DE FIGURAS

1.1	Docagem Molecular	13
2.1	Visualizador View	15
2.2	Visualizador PyMOL	16
2.3	Visualizador POSMOL	17
2.4	AutoDock Vina	17
2.5	Ligand docking and binding	18
2.6	AUDocker LE	19
4.1	Ilustração do menu movement	21
4.2	Exemplo de Redes biológicas	22
4.3	Exemplo de molécula	23
4.4	Exemplo do modo surface	24
4.5	Exemplo de Fieldlines	24
4.6	Hiperboloide de Uma Folha.	25
4.7	Hiperboloide de Duas Folhas	25
4.8	Representação das HyperBalls	26
5.1	Disambiguation Canvas passo a passo: (a) usuário aponta para área onde se encontra o objeto alvo; (b) ao executar o toque no tela os objetos são rearranjados do volume para o retângulo na tela; (c) o retângulo é mapeado absolutamente para o touchscreen do smartphone, o usuário desliza o polegar com o objetivo de apontar o objeto alvo.	27
5.2	Layouts padrão para área útil do touchscreen propostas para o Dcanvas. . .	28
6.1	Comunicação Cliente-Servidor.	30
6.2	Ilustrando Volume de Seleção.	31
6.3	Ilustrando Plano de Seleção.	32
6.4	Linha e Glow na Seleção do Plano.	33
6.5	Ilustração dos movimentos do smartphone	34
7.1	Exemplo de área selecionada para Docagem molecular.	36

LISTA DE TABELAS

6.1	Manipulação da Molécula	33
-----	-----------------------------------	----

LISTA DE ABREVIATURAS E SIGLAS

Dcanvas Disambiguation Canvas

Umol Unitymol

PDB Protein Data Bank

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Motivação	12
1.2	Objetivos	13
1.3	Contribuição	13
1.4	Estrutura desse trabalho	13
2	TRABALHOS RELACIONADOS	15
3	PROBLEMAS DE SELEÇÃO EM AMBIENTES DESORDENADOS	20
3.1	Precisão	20
3.2	Ambiguidade	20
3.3	Complexidade	20
4	UNITYMOL	21
4.1	Versões	21
4.2	Acessando o Banco de Dados	22
4.3	Métodos Gráficos	22
4.4	Usando GPU shaders nas Hypeballs	23
5	DISAMBIGUATION CANVAS	27
5.1	Técnicas de volume de seleção	27
5.2	Mapeando objetos na tela (canvas)	27
5.3	Contribuição	28
6	INTEGRAÇÃO DCANVAS E UMOL	29
6.1	Parte Gráfica	29
6.2	Implementação	29
6.2.1	Apontador	29
6.2.2	Volume de Seleção	30
6.2.3	Objetos no Plano	30
6.2.4	Seleção	31
6.2.5	Movimento da Molécula	32
6.3	Um Exemplo de Uso	32
7	ESTUDO DE CASO - DOCAGEM MOLECULAR	35
7.1	Algoritmo de Docagem	35
7.1.1	Algoritmo de Busca	35
7.1.2	Função de Pontuação	35

7.2	Busca Manual	35
7.3	Teste	36
8	CONCLUSÕES E TRABALHOS FUTUROS	37
	REFERÊNCIAS	38

1 INTRODUÇÃO

A Química é uma ciência alicerçada em um paradigma funcional-estruturalista que adota o princípio fundamental de que as propriedades e transformações dos materiais podem ser satisfatoriamente explicadas pela teoria atômico-molecular. No estágio atual, a representação de átomos, moléculas, adutos, agregados e outras partículas constituintes dos materiais são fatores determinantes para a elaboração do pensamento químico e esta é uma das razões para adotar a visualização de objetos moleculares como estratégia de ensino, especialmente quando já é possível levá-la à sala de aula por meio de computadores disponíveis em grande parte das escolas (GIORDAN et al., 2004). O visualizador e editor de moléculas Unitymol(Umol) propõe uma maneira fácil e rápida para visualização de moléculas, por meio de uma engine de games, Unity 3D (LV et al., 2013). Este possibilita que a molécula seja manipulada por meio de rotações. Nós acrescentamos mais um meio de interação com a moléculas, a seleção.

Seleção é uma das quatro formas fundamentais de interação em mundos virtuais 3D (BOWMAN et al., 2004) e consiste em especificar objetos no ambiente virtual para ações subsequentes(STEED, 2006). A seleção em ambientes complexos apresenta problemas que estão ainda em aberto. Devido ao grande número de objetos, há dificuldade na seleção precisa, posto que há muitos casos de oclusão entre objetos.

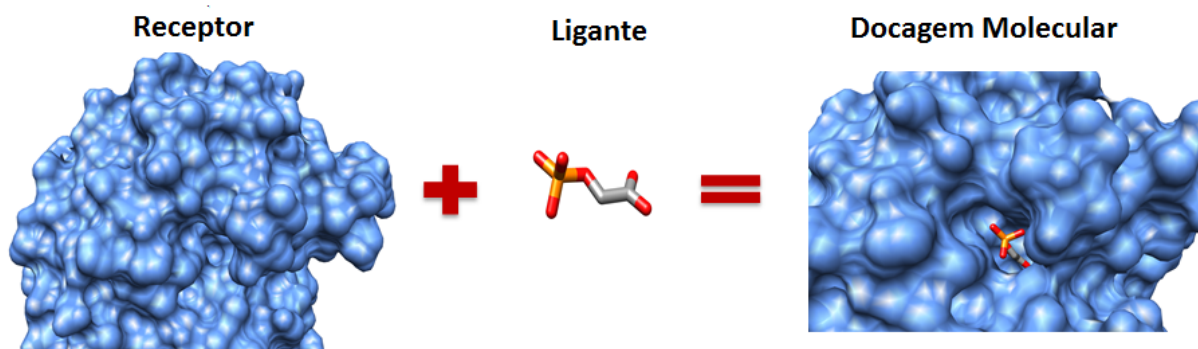
Disambiguation Canvas(Dcanvas) é uma técnica de seleção refinada progressiva que consiste na definição de um subconjunto de objetos na primeira fase, seguido por uma fase de desambiguação. Essa técnica propõe uma maneira rápida e precisa para selecionar objetos em ambientes densos. Por conseguinte, a técnica Dcanvas é apropriada para realizar seleção e manipulação de moléculas no editor Umol. Este trabalho propõe a integração entre os dois sistemas citados anteriormente.

Com ambos os sistemas integrados, propõe-se uma maneira de executar manualmente a primeira etapa em um algoritmo de docagem molecular.

1.1 Motivação

A Docagem Molecular, também conhecida como Acoplamento Molecular, Ancoragem Molecular, ou *Docking*, no campo da modelagem é um método que prevê a orientação preferencial de uma molécula (ligante) a uma segunda (receptor)(figura 1.1), quando ligados entre si, para formar um complexo estável. O conhecimento da orientação preferida, por sua vez, pode ser utilizado para prever a força de associação ou a afinidade de ligação entre duas moléculas. Atualmente, existem algoritmos genéticos capazes de executar a docagem molecular, mas esses analisam a molécula inteira em busca de uma área na qual as moléculas possam se acoplar. Uma maneira utilizada para diminuir esse tempo é a seleção manual por um especialista, da área preferencial na qual o *docking* pode ocorrer. Com o auxílio da técnica Dcanvas esse trabalho propõe uma maneira rápida e precisa para se executar a etapa manual do *docking*.

Figura 1.1 – Docagem Molecular



Fonte: Intech

1.2 Objetivos

A integração dos dois sistemas (Dcanvas e Umol) visa uma melhor aproximação para a interação no visualizador Umol, uma vez que muitas das manipulações feitas com a molécula, como rotações e translações, são realizadas com o mouse e algumas das rotações mais específicas são realizadas no menu de interação (figura 4.1). Além disso, como já foi apresentado acima, uma maneira mais rápida para selecionar a área preferencial para o *docking* molecular.

1.3 Contribuição

As principais contribuições deste trabalho são:

- Uma nova maneira de interagir no visualizador Umol
- A possibilidade de selecionar átomos
- Uma proposta para diminuir o tempo do algoritmo de docagem molecular

1.4 Estrutura desse trabalho

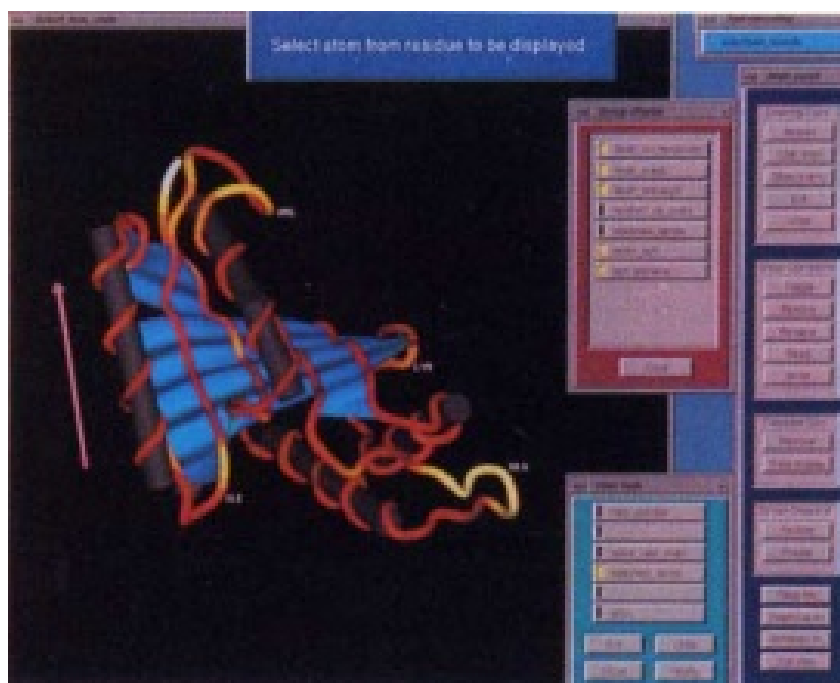
O restante deste trabalho está organizado da seguinte maneira. Para contextualizar o leitor, o capítulo 2 lista alguns trabalhos relacionados, ilustrando o que mais está sendo desenvolvido na área. O capítulo 3 aborda alguns dos principais problemas de seleção, os quais pretendemos solucionar com a nossa aplicação. O capítulo 4 apresenta o editor de moléculas Umol, descrevendo sua interface e suas funcionalidades. O capítulo 5 apresenta a técnica de seleção Dcanvas e seu funcionamento. O capítulo 6 descreve como foi implementada a integração entre os dois sistemas citados anteriormente e suas adaptações para o seu melhor funcionamento. O capítulo 7 apresenta a Docagem Molecular, a tarefa que será aplicada ao nosso sistema. Tam-

bém descreve o teste que pretendemos realizar com especialistas. O capítulo 8 apresenta as conclusões desse trabalho e nos destacamos nossas sugestões para futuros desenvolvimentos.

2 TRABALHOS RELACIONADOS

Neste capítulo vamos apresentar os trabalhos semelhantes que foram/estão sendo desenvolvidos na área. Iniciamos citando o VIEW (BERGMAN et al., 1993) é um visualizador para estudar estruturas moleculares. A visualização é construída através de outras ferramentas de desenho em uma biblioteca. Nessa ferramenta o usuário pode definir, por meio do mouse, a sequência de interação de acordo com sua preferência. Por exemplo, durante a geração da molécula, o usuário pode escolher qual parte quer que cada ferramenta presente na biblioteca desenhe, assim como pode escolher qual transformação vai executar em qual parte. Por exemplo, rotacionar e depois transladar (figura 2.1). Citamos também o PyMOL (DELANO, 2002),

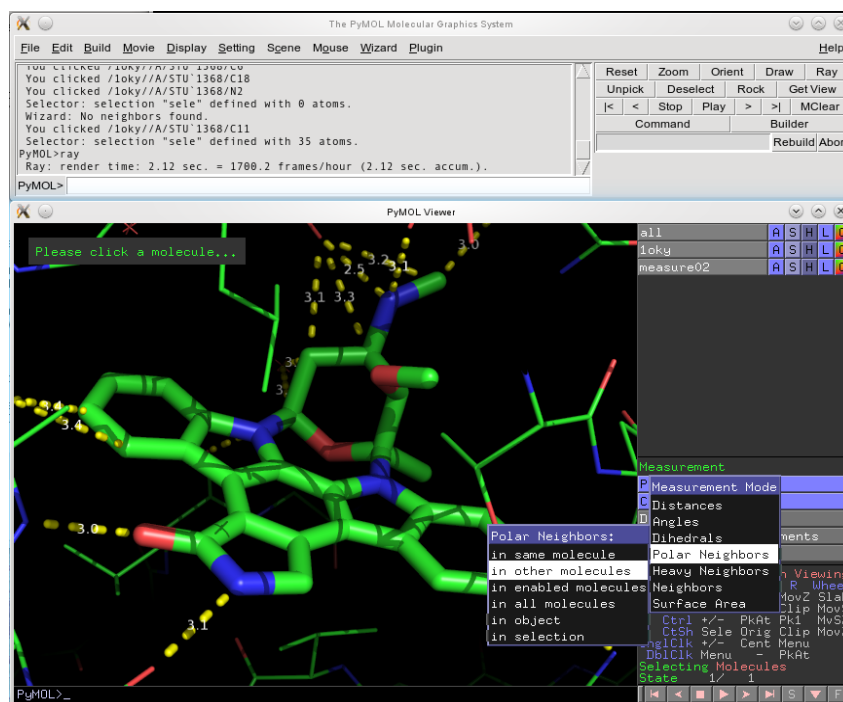
Figura 2.1 – Visualizador View



Fonte: (BERGMAN et al., 1993)

um dos visualizadores de moléculas mais utilizados para pesquisa e estudos. PyMOL possui um interno *ray caster* para *ray tracing*, técnica que aumenta as sombras e suaviza as bordas para renderizar imagens 3D de alta resolução. A interação no visualizador é realizada via cliques de mouse e/ou menus de contexto. Também pode receber entradas via mouse, scripts ou linha de comando (figura 2.2). Outro visualizador que podemos citar é o POSMOL (LEE; CHUNG; KIM, 2004) o qual é capaz de desenhar, renderizar, editar e analisar moléculas. A interação acontece por meio de *toolbars*, nas quais é possível transladar, rotacionar e escalar as moléculas (figura 2.3). Também podemos apresentar trabalhos que abordam docagem molecular, como por exemplo o **AutoDock Vina** (TROTT; OLSON, 2010), ferramenta de docagem muito utilizada para pesquisa, com uma teoria própria para fazer a predição da área preferencial onde o

Figura 2.2 – Visualizador PyMOL

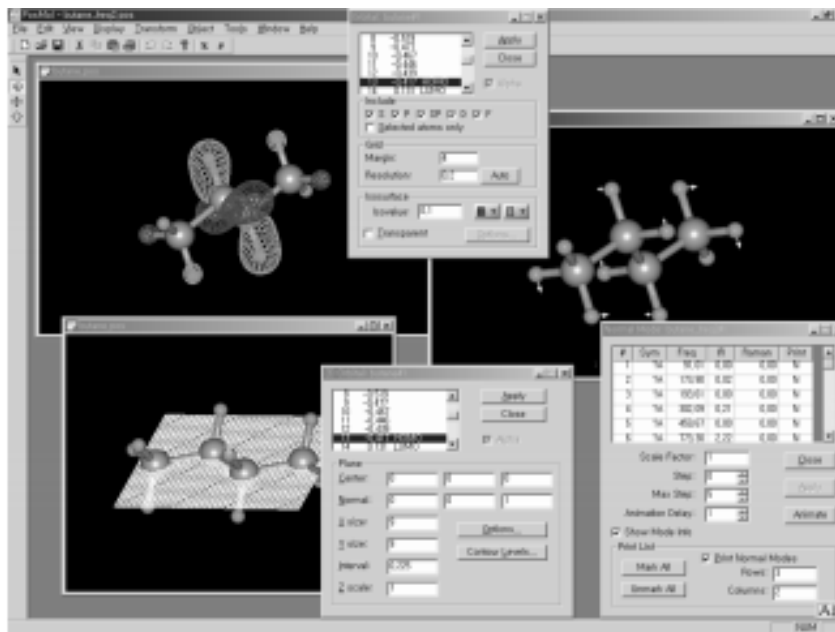


Fonte: www.pymol.org

ligante pode se acoplar ao receptor (Docagem Molecular) e depois aplicar sua função de pontuação (figura 2.4). Também temos o **Ligand docking and binding site analysis with PyMOL and Autodock/Vina** (SEELIGER; GROOT, 2010) que usa o PyMOL como sistema gráfico. O Autodock and Vina, é um programa open-source, desenvolvido para executar docagem molecular. Além desses, também citamos o AUDocker LE (SANDEEP et al., 2011), que nada mais é do que uma interface gráfica para o Autodock com o intuito de possibilitar uma interação mais acessível para seus usuários (figura 2.6).

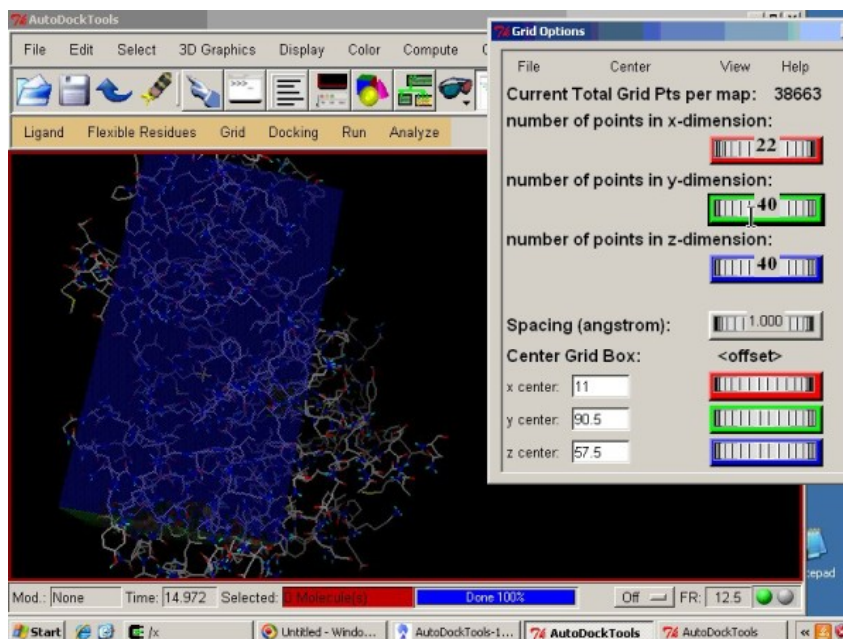
Como podemos observar nos trabalhos acima, a maioria usa métodos tradicionais de interação, mouse e menus, e a docagem molecular é realizada em duas partes principais. Primeiro uma heurística para encontrar a área preferencial e depois uma função de pontuação para executar a docagem. Neste trabalho nós propusemos uma interação mais rápida e intuitiva com o uso de um smartphone e uma alternativa manual para a etapa de busca na docagem molecular.

Figura 2.3 – Visualizador POSMOL



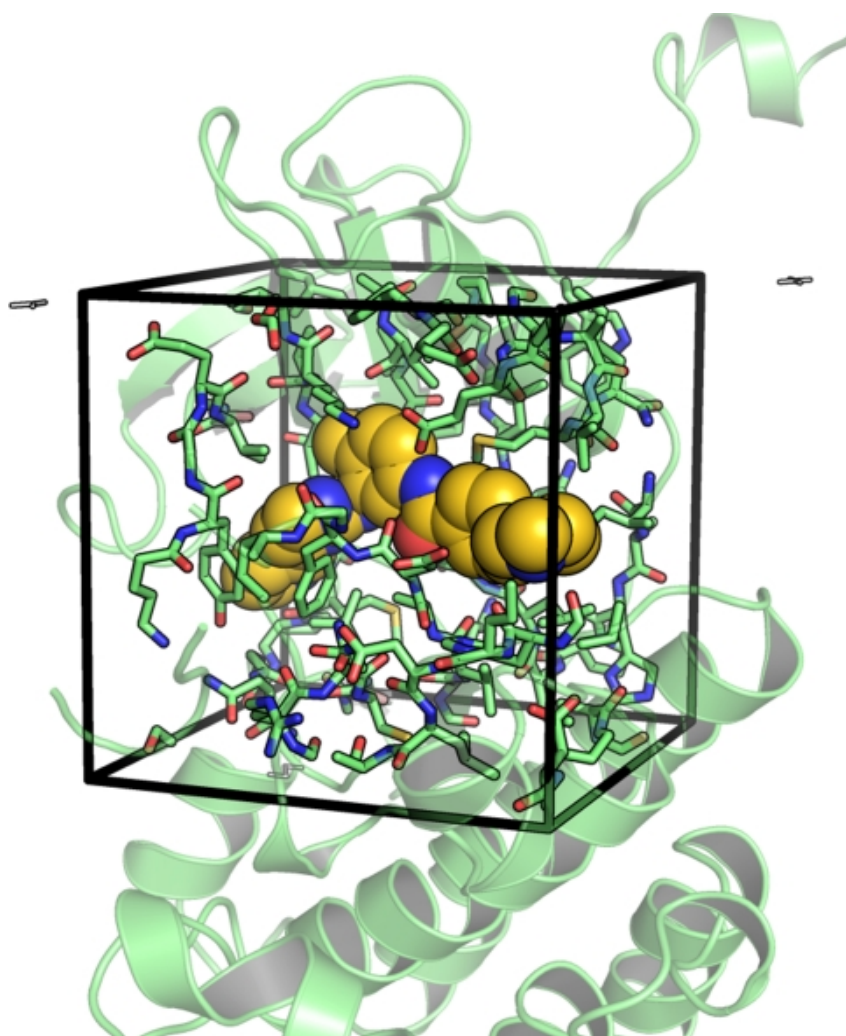
(LEE; CHUNG; KIM, 2004)

Figura 2.4 – AutoDock Vina



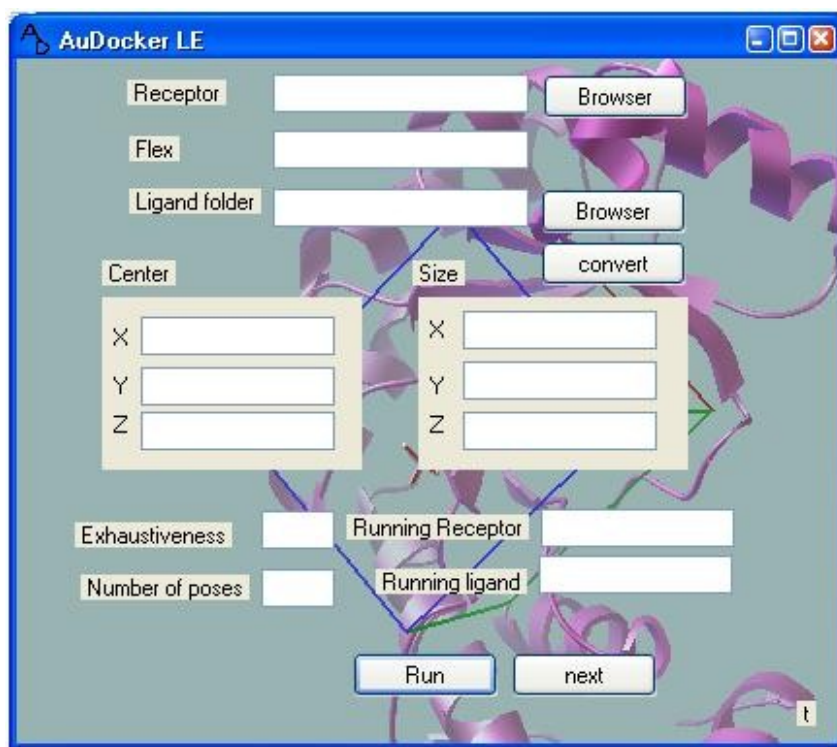
(TROTT; OLSON, 2010)

Figura 2.5 – Ligand docking and binding



(SEELIGER; GROOT, 2010)

Figura 2.6 – AuDocker LE



(SANDEEP et al., 2011)

3 PROBLEMAS DE SELEÇÃO EM AMBIENTES DESORDENADOS

Ambientes virtuais, principalmente os densos/complexos e com muitos objetos, apresentam três grandes problemas de seleção: precisão, ambiguidade e complexidade.

3.1 Precisão

Um dos maiores problemas para seleção com apontamento é a precisão. Em grandes displays esse problema acentua-se ainda mais. Os dispositivos de entrada normalmente usados nessas técnicas possuem 6 graus de liberdade, três, responsáveis pela posição do dispositivo, são usados para posicionar a origem do raio e os outros três, responsáveis pela orientação do dispositivo, são usados para definir a direção do raio. Como um pequeno movimento angular com o apontador resulta em movimentos amplificados no ponto de intersecção do raio, o controle de orientação é predominante sobre o controle de posição em seleção por apontamento e quanto maior a distância, maior a amplificação do movimento. O tamanho do objeto a ser selecionado influencia diretamente na dificuldade da seleção com apontadores.

Métodos mais tradicionais para seleção, como mouse, por exemplo, apresentam uma precisão maior. Entretanto, em ambientes muito desorganizados pode ocorrer oclusão, o que dificulta a seleção do objeto alvo, ou pode tornar-se uma tarefa demorada e cansativa, em grandes displays, por exemplo (DEBARBA, 2012).

3.2 Ambiguidade

A ambiguidade ocorre quando o sistema não é capaz de distinguir qual objeto dentre um grupo de objetos o usuário deseja selecionar. No caso de uma seleção com volume, mais de um objeto pode estar contido no volume usado e dessa maneira objetos não alvos serão selecionados juntos.(DEBARBA, 2012)

3.3 Complexidade

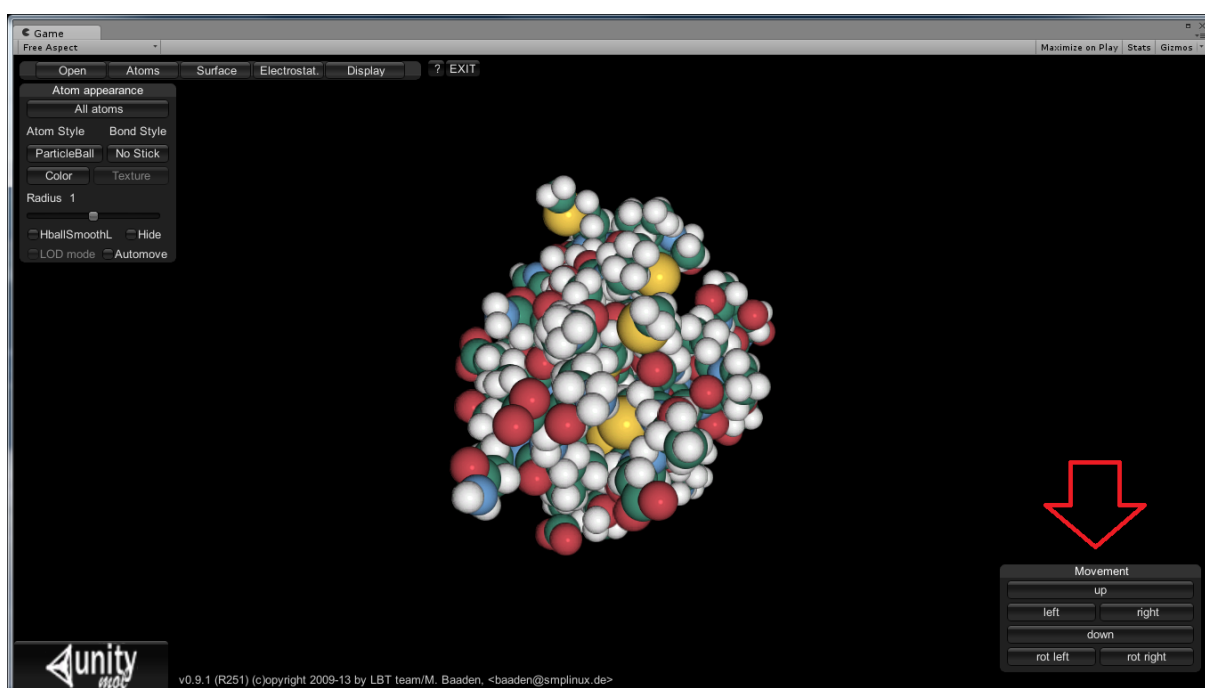
A complexidade diz respeito a objetos que são animados, ou seja, que mudam sua posição com o tempo. Objetos que também aparecem por um tempo limite no display também tem uma complexidade maior para serem selecionados. Ainda podemos levar em conta objetos que mudam de forma ou de tamanho com o tempo (DEBARBA, 2012).

4 UNITYMOL

Unitymol é um editor e visualizador de moléculas (figura 4.3) e redes biológicas (figura 4.2) desenvolvido na *game engine* Unity 3D. O editor permite carregar moléculas do *Protein Data Bank (PDB)*, ou em português Banco de Dados de Proteínas, que com a colaboração de bioquímicos e químicos de todo o mundo armazena milhares de modelos 3D de moléculas e ácidos nucleicos.

O editor também permite a manipulação das moléculas, permitindo ao usuário executar movimentos para cima, para baixo, para esquerda e para direita, os quais podem ser executados com o mouse, ou no menu *movement* e rotações para a esquerda e para a direita, as quais só podem ser executadas no menu *movement* (na figura 4.1 exemplo do menu). O Unitymol também é capaz de apresentar as moléculas no modo *surface* como na figura 4.4 e *animated electrostatic field lines* conforme figura 4.5.

Figura 4.1 – Ilustração do menu movement

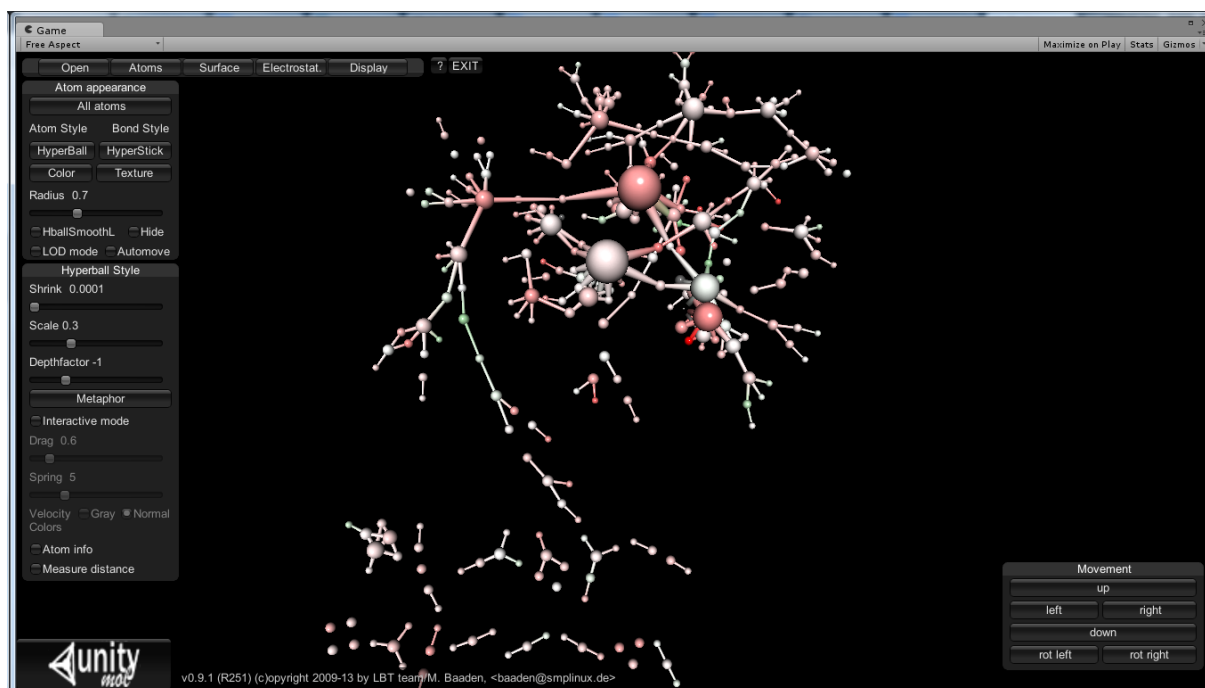


Fonte: Printscreen Umol

4.1 Versões

O Unitymol é desenvolvido em duas versões: uma versão desktop e uma versão web. Ambas possuem código aberto que está disponibilizado no sourceforge(<http://sourceforge.net/projects/unitymol/>).

Figura 4.2 – Exemplo de Redes biológicas



Fonte: Printscreen Umol

4.2 Acessando o Banco de Dados

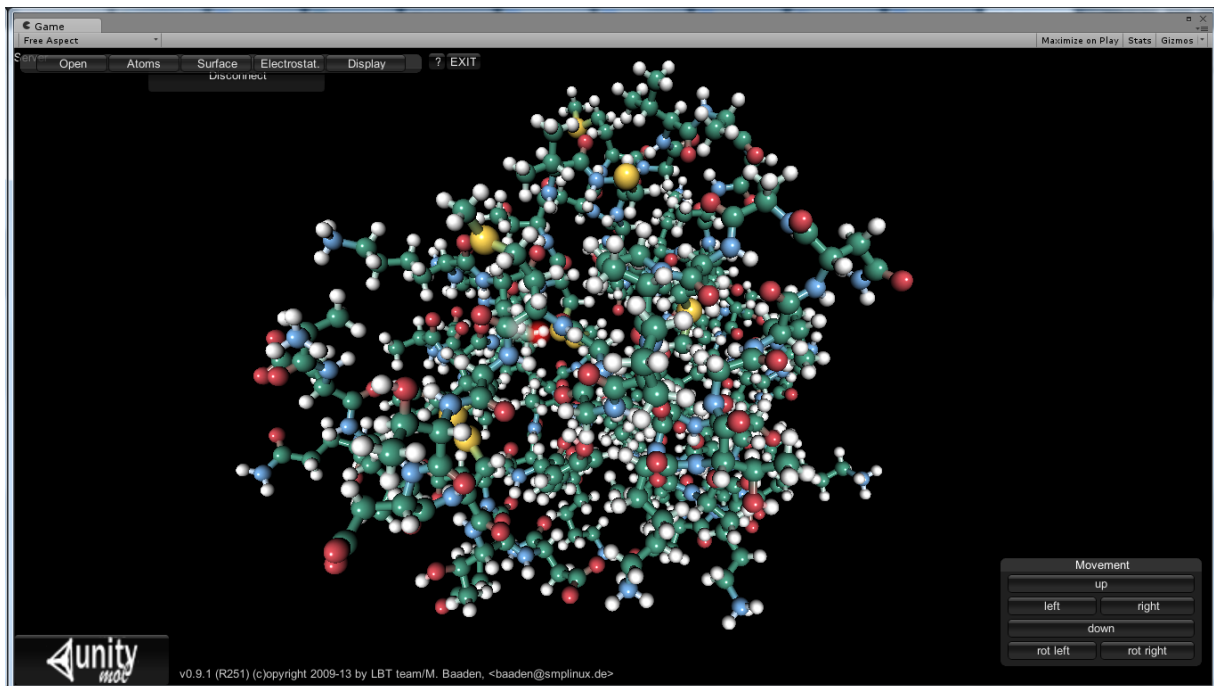
Os dados estruturais são carregados a partir do formato PDB (BERMAN et al., 2000). Os arquivos podem ser importados do disco local ou podem ser baixados diretamente do *PDB server* por meio do ID da molécula. Devido a configurações de segurança da versão Web da Unity3D, os arquivos precisam ser carregados por meio de um servidor intermediário. Essa restrição não se aplica à versão desktop.

A topologia molecular é construída usando distâncias interatômicas típicas para detectar limites. As superfícies do modo *surface* são construídas, ou externamente como *meshes* no formato OBJ, ou internamente usando a aproximação *density grid isosurface*. Os potenciais eletrostáticos são computados pelo APBS (*Adaptative Poisson-Boltzmann Solver*, um programa grátis para calcular eletrostática macromolecular). e as redes biológicas são representadas usando *extensible Graph Markup and Modelling Language (XGMML)*.

4.3 Métodos Gráficos

Para a representação gráfica das moléculas foram usados, além das primitivas da Unity3D, cubo e esfera, uma representação de HyperBalls baseada em raycast. Esferas e cubos representam os átomos nas moléculas e os nodos em redes biológicas. Ligações entre eles são represen-

Figura 4.3 – Exemplo de molécula



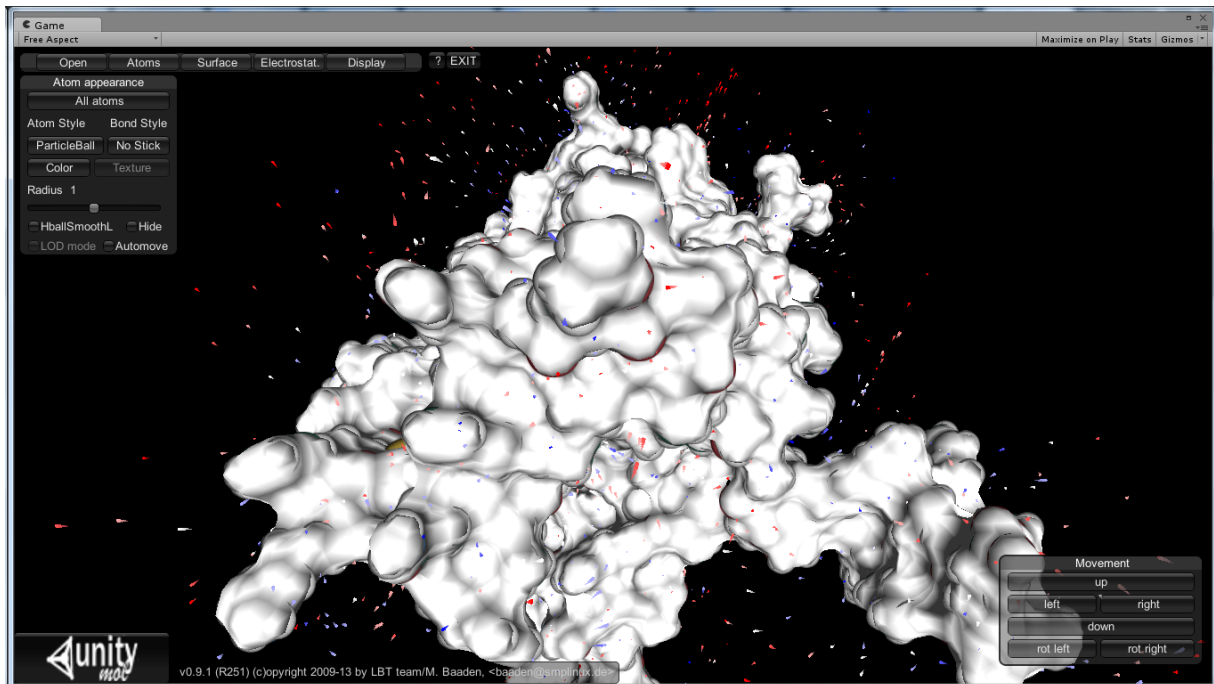
Fonte: Printscreen Umol

tadas usando linhas.

4.4 Usando GPU shaders nas Hypeballs

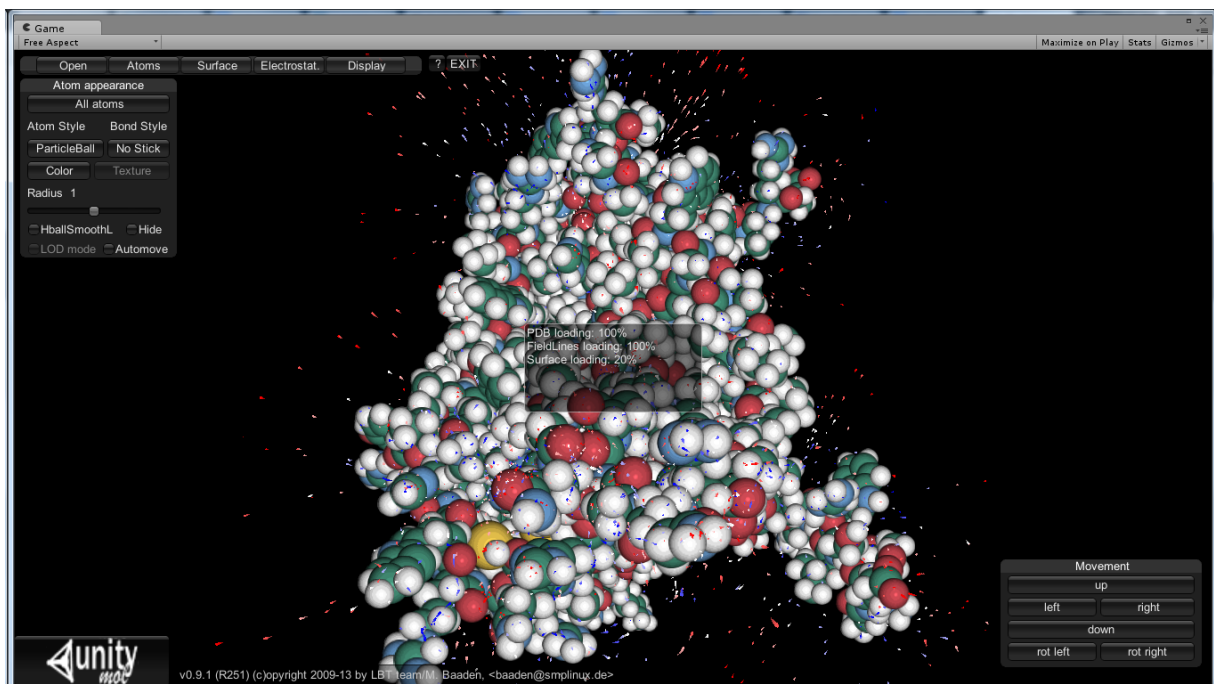
Para representar os átomos foi desenvolvida uma representação chamada *HyperBalls*. Essa representação se vale da capacidade de programar as placas gráficas e usa um algoritmo de *raycasting* de alta qualidade com um bom desempenho para exibição. Essa representação é composta por esferas representando átomos conectados por hiperbolóides. Usando os hiperbolóides, a representação pode ser adaptada para ilustrar a evolução contínua da ligação passando de hiperbolóide de uma folha (figura 4.6), para hiperbolóide de duas folhas (figuras 4.7).

Figura 4.4 – Exemplo do modo surface



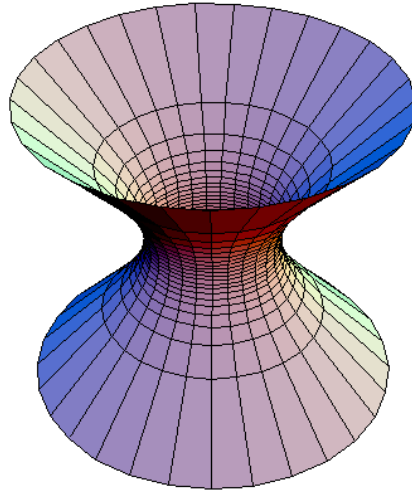
Fonte: Printscreen Umol

Figura 4.5 – Exemplo de Fieldlines



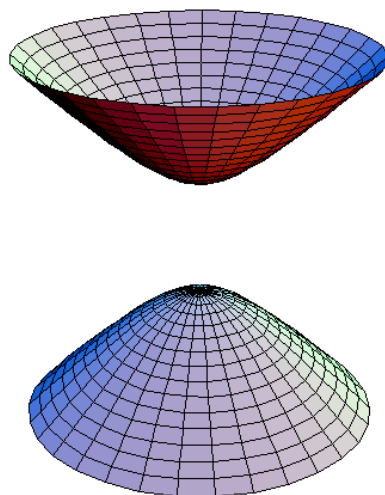
Fonte: Printscreen Umol

Figura 4.6 – Hiperboloide de Uma Folha.



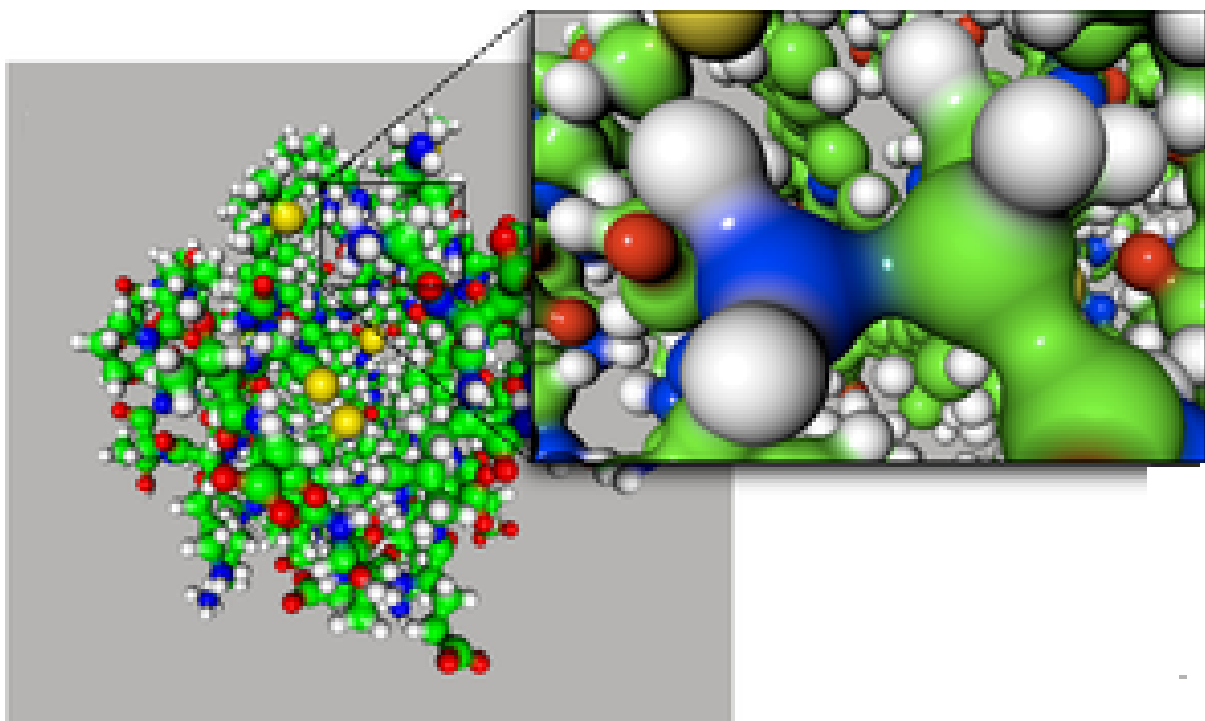
Fonte:Wikipedia

Figura 4.7 – Hiperboloide de Duas Folhas



Fonte:Wikipedia

Figura 4.8 – Representação das HyperBalls

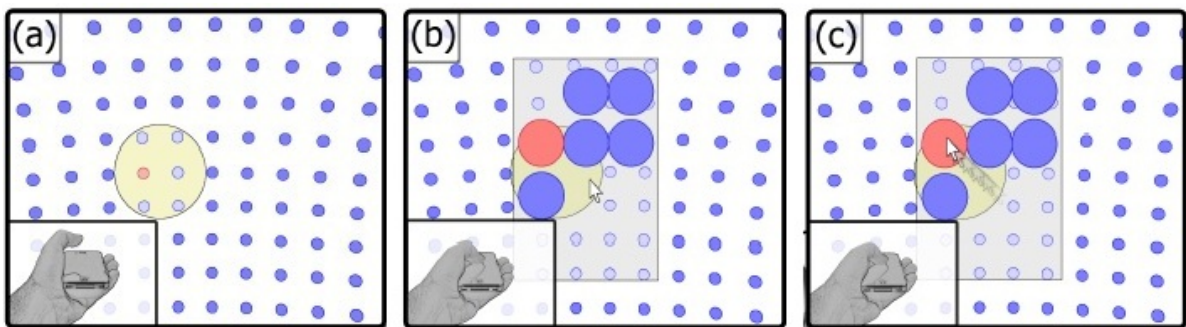


Fonte: (LV et al., 2013)

5 DISAMBIGUATION CANVAS

Disambiguation Canvas é uma técnica de seleção em ambientes 3D por refinamento progressivo usando um dispositivo móvel (smartphone) e é composta por dois passos. No primeiro, o usuário define um subconjunto de objetos por meio de uma técnica de apontamento como um volume de seleção. Usando um smartphone, executa um toque com o polegar no touchscreen e dá início ao próximo passo. O passo subsequente consiste na desambiguação do alvo desejado entre os objetos pertencentes ao subconjunto previamente selecionado. Isso é realizado usando o touchscreen do smartphone. Nessa etapa, um retângulo contendo o subconjunto de objetos a ser desambiguado aparece na tela. Esse retângulo é mapeado 1:1 com a tela do smartphone. Deslizar o polegar sobre a touchscreen permite a sobreposição do cursor com o objeto desejado. O gesto de levantar o polegar do touchscreen executa a seleção (DEBARBA, 2012).

Figura 5.1 – Disambiguation Canvas passo a passo: (a) usuário aponta para área onde se encontra o objeto alvo; (b) ao executar o toque na tela os objetos são rearranjados do volume para o retângulo na tela; (c) o retângulo é mapeado absolutamente para o touchscreen do smartphone, o usuário desliza o polegar com o objetivo de apontar o objeto alvo.



Fonte: (DEBARBA, 2012)

5.1 Técnicas de volume de seleção

As técnicas mais comuns para volumes de seleção utilizam cone e esferas. O Disambiguation Canvas suporta ambos os volumes na sua implementação.

5.2 Mapeando objetos na tela (canvas)

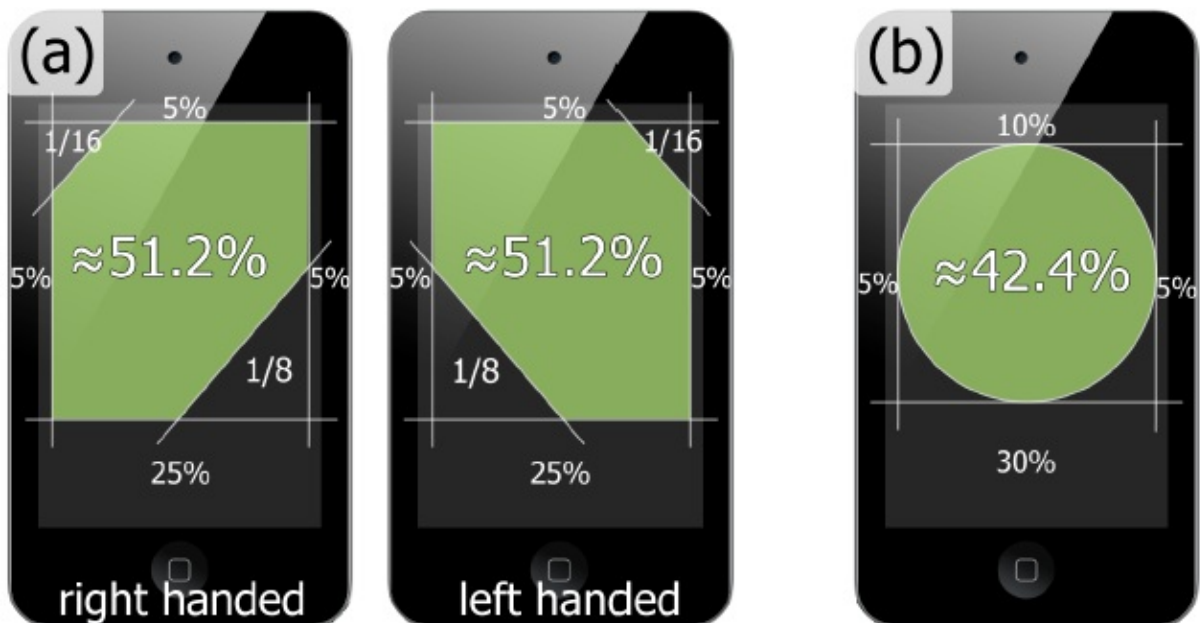
Quando inicia-se o segundo passo do Dcanvas, o subconjunto de objetos deve ser reorganizado lado a lado em um plano na tela. Como a maioria dos usuários não alcança toda a área da tela do smartphone, a técnica propõe a redução da área útil na qual os objetos serão reorganizados. Dessa maneira os objetos são trazidos ao alcance dos usuários. Foram implementados dois

layouts: o primeiro consiste em aproximadamente 51% da área total do touchscreen e tem um *layout* específico para destros e outro para canhotos (figura 5.2(a)). O segundo *layout* consiste em aproximadamente 42% da área total do touchscreen (figura 5.2(b)) (DEBARBA, 2012).

5.3 Contribuição

A técnica Dcanvas tem como grande contribuição ser rápida e precisa em apenas dois passos. Além disso é capaz de solucionar boa parte dos problemas apresentados no capítulo 3. Com essa técnica podemos selecionar precisamente um objeto no plano com o uso do touchscreen do smartphone e toda a técnica em si serve para desambiguar objetos na seleção, portanto foi possível solucionar o problema da precisão e da ambiguidade.

Figura 5.2 – Layouts padrão para área útil do touchscreen propostas para o Dcanvas.



Fonte: (DEBARBA, 2012)

6 INTEGRAÇÃO DCANVAS E UMOL

Com o objetivo de melhorar a interação no editor de moléculas Umol, nós propusemos a sua integração com a técnica DCanvas. Além de usarmos um smartphone para realizar os movimentos que tradicionalmente eram executados com o mouse, ou no menu *movement*, permitimos agora ao usuário selecionar átomos na molécula.

6.1 Parte Gráfica

Usamos as próprias primitivas da Unity3D para a representação gráfica da aplicação. Para representar o volume de seleção usamos um wireframe esférico (ver figura 6.2). Um cilindro rotacionado para o cursor de seleção no plano e um plano transparente ((ver figura 6.3) para não confundir os átomos no plano de desambiguação, com os átomos na molécula.

6.2 Implementação

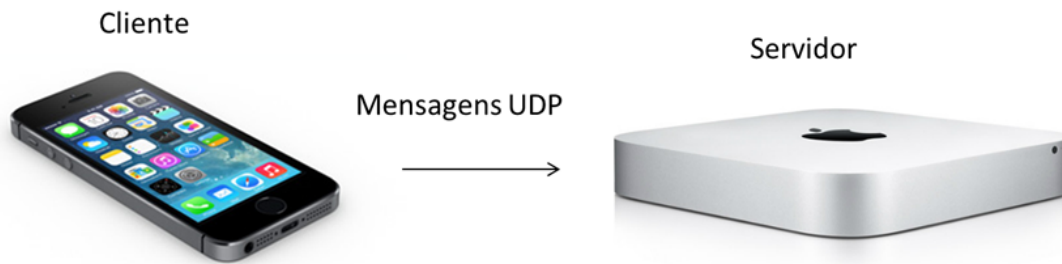
Este trabalho foi desenvolvido na game engine Unity3D em C#, principalmente porque o visualizador Umol foi desenvolvido nessa engine. Entretanto, é importante ressaltar como as ferramentas criadas para desenvolvimento de jogos vem sistematicamente sendo cada vez mais usadas para estudo e pesquisa. A alta performance que as engines podem atingir hoje em dia e o grande nível de abstração que elas proporcionam garantem um ambiente propício para o desenvolvimentos de trabalhos científicos. A aplicação foi desenvolvida em um Mac Mini Intel Core I5 dual core de 1,4GHz com 4Mb de memória ram e um iPhone 5S.

A integração foi desenvolvida como um sistema cliente-servidor com trocas de mensagens usando o protocolo UDP(figura 6.1). O smartphone atua como cliente ao passo que a aplicação desktop faz o papel de servidor. A aplicação Dcanvas foi desenvolvida no mesmo sistema. Entretanto, o cliente foi desenvolvido em objective C e a aplicação servidor foi desenvolvida na Ogre 3D. Portanto, o sistema foi totalmente reimplementado na Unity3D.

6.2.1 Apontador

Como citado acima, o cliente da nossa aplicação é o smartphone, dispositivo que usamos como apontador para controlar o volume de seleção. O celular envia as informações de rotação baseada no *sensor fusion* do próprio celular (tanto iOS quanto Android possuem sensor fusion, não foi testado em windows phone). O *sensor fusion* é a combinação dos dados dos três sensores presentes nos smartphones atuais: acelerômetro, magnetômetro e giroscópio para melhorar a performance do sistema. A informação de rotação enviada pelo smartphone é literalmente um quaternion. A posição do apontador é calculada da seguinte maneira: criamos um plano que funciona como se fosse a posição na tela para onde desejamos apontar, então usamos um vetor

Figura 6.1 – Comunicação Cliente-Servidor.



Fonte: Autor

à uma distância arbitrária apontando para o centro desse plano. Nesse vetor é aplicada a rotação do quaternion. Em seguida, disparamos um *raycast* na direção desse vetor e calculamos o ponto de intersecção do *raycast* com o plano e dessa maneira temos a posição X,Y que estamos apontando na tela.

6.2.2 Volume de Seleção

A técnica Dcanvas permite o uso de esferas e cones para o volume de seleção. No nosso caso, usamos uma esfera. Para verificar a intersecção com os objetos, átomos no caso do Umol, usamos a primitiva *Collider* da Unity3D. A própria engine contém funções que identificam se um objeto está contido ou não em um volume. Todos os átomos que entram no *Collider*, usamos a função *OnColliderEnter*, são armazenados em uma lista. Assim que deixam o *Collider*, função *OnColliderExit*, são retirados da lista. Dessa maneira, temos controle sobre o subconjunto de átomos que está dentro do volume de seleção.

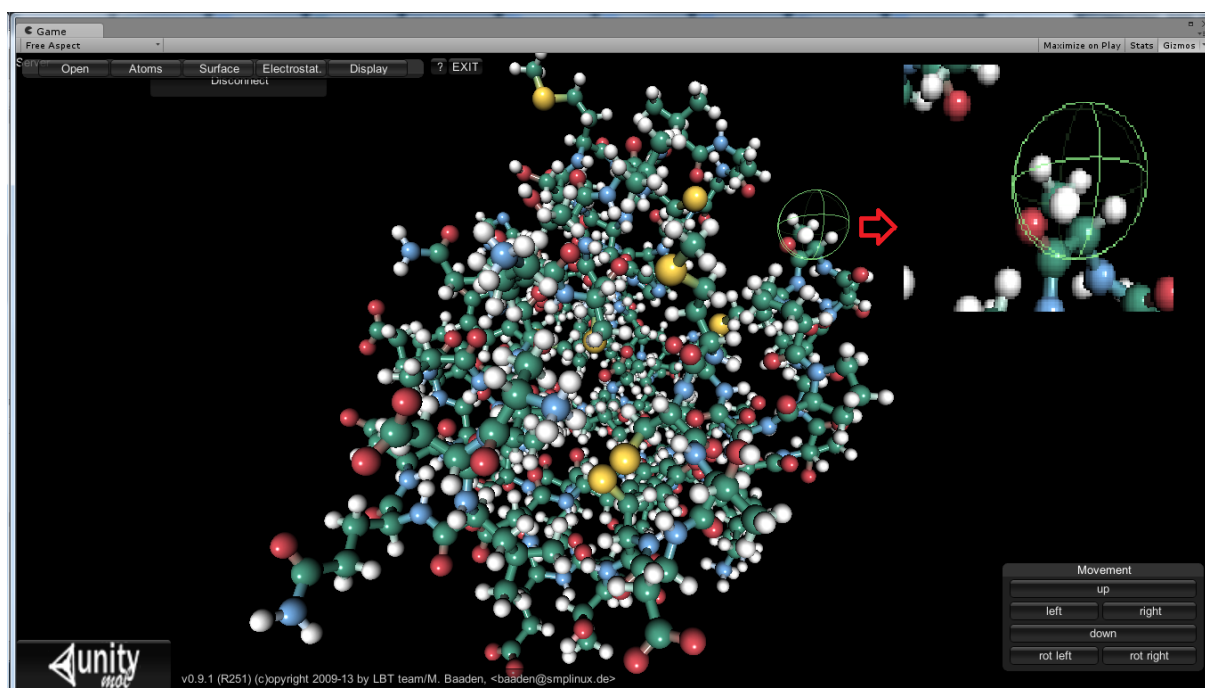
6.2.3 Objetos no Plano

Para a seleção no plano escolhemos o segundo *layout* (figura 5.2(b)), pois é apropriado para qualquer pessoa, independente da mão preferencial. Antes de dispor os objetos no plano é calculado um *grid* no qual os objetos serão alocados. Esse *grid* é calculado de acordo com o valor \mathcal{N} , número de objetos do subconjunto presente no volume de seleção. O cálculo é feito da seguinte maneira: extraímos a raiz de \mathcal{N} , se a raiz for exata construímos um *grid* $\mathcal{N}\mathcal{X}\mathcal{N}$. Se a raiz não for exata, somamos 1 ao valor de \mathcal{N} e então construímos um *grid* de tamanho $\mathcal{N}+1\mathcal{X}\mathcal{N}+1$.

Os objetos são escalados novamente para encaixar no *grid* e manter o mapeamento 1:1 com o touchscreen do smartphone. Quanto mais objetos, menor eles serão no plano. Entretanto, como temos objetos de tamanhos diferentes, os objetos são escalados em relação ao seu tamanho,

para assim, mantermos a proporção. Para evitar a ambiguidade definimos um raio $\mathcal{R} = 5$ para o volume esférico de seleção e com isso temos um \mathcal{N} máximo médio de aproximadamente 13, dessa maneira nosso grid na maioria das vezes tem tamanho 16 (4x4). Se usarmos um raio $\mathcal{R} > 5$ o grid chega ao tamanho 36 (6x6), tornando os objetos no plano menores e, conseqüentemente, mais difíceis de serem selecionados, por isso optamos por $\mathcal{R} = 5$.

Figura 6.2 – Ilustrando Volume de Seleção.

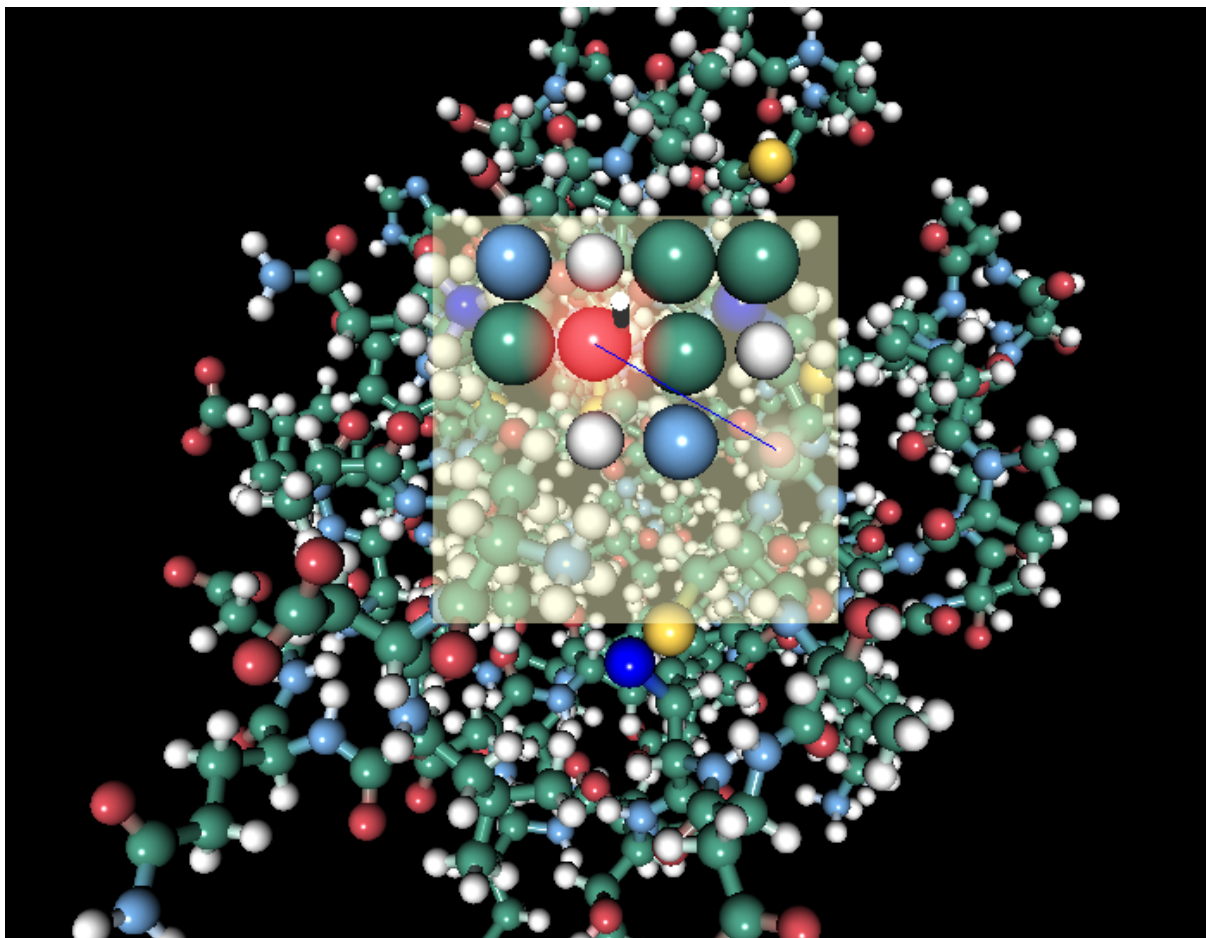


Fonte: Printscreen Umol

6.2.4 Seleção

Para garantir a integridade da molécula durante a seleção, são instanciados novos objetos referentes aos átomos originais. Os novos átomos são atribuídos como filhos dos originais. Desse modo, no plano para desambiguação estão os átomos filhos que, ao serem selecionados, referenciam os pais, átomos originais, na molécula. Os átomos presentes no plano são sócias dos átomos na molécula, ou seja, são uma duplicada dos originais, criados para evitar a perda do contexto, selecionar o átomo *A* significa selecionar o mesmo átomo na molécula. Para evitar ambiguidade no momento em que o cursor intersecta o átomo filho, é aplicado um *glow* da cor branca, enquanto em seu pai é aplicado um *glow* de cor vermelha. Ainda para reforçar esse aspecto, no momento da intersecção do cursor com o átomo filho, uma linha é renderizada conectando o pai com o filho, para que desse maneira, o usuário não tenha dúvidas ao selecionar o objeto alvo (figura 6.4).

Figura 6.3 – Ilustrando Plano de Seleção.



Fonte: Printscreen Umol

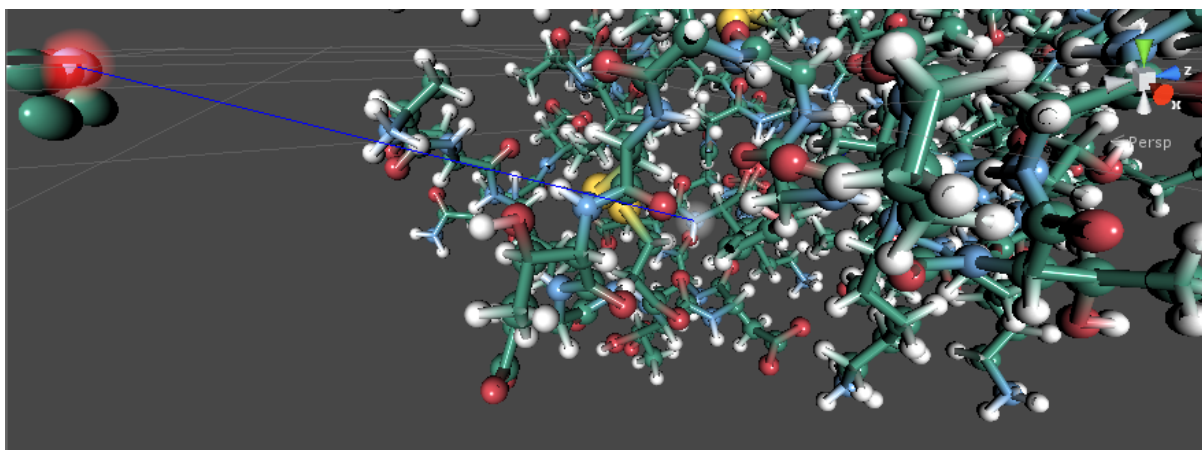
6.2.5 Movimento da Molécula

As manipulações que podem ser feitas na molécula no Umol, também foram implementadas para serem executadas com o smartphone. Escolhemos o acelerômetro, pois tem 6 graus de liberdade e com isso é possível executar todos os movimentos que originalmente são realizados usando o mouse ou o menu *moviment*. Usando os movimentos de pitch, yaw e roll (figura 6.5), podemos mover a molécula para cima e para baixo, para direita e para esquerda, e realizar rotação para direita e para esquerda também, conforme a tabela 6.1.

6.3 Um Exemplo de Uso

Vamos supor que um usuário deseja selecionar um átomo de uma molécula para um determinado fim. Primeiramente, ele carrega a molécula desejada no visualizador Umol. Então, inicia o servidor na sua aplicação desktop. Em seguida, inicia o cliente em seu smartphone. Depois

Figura 6.4 – Linha e Glow na Seleção do Plano.



Fonte: Printscreen Umol

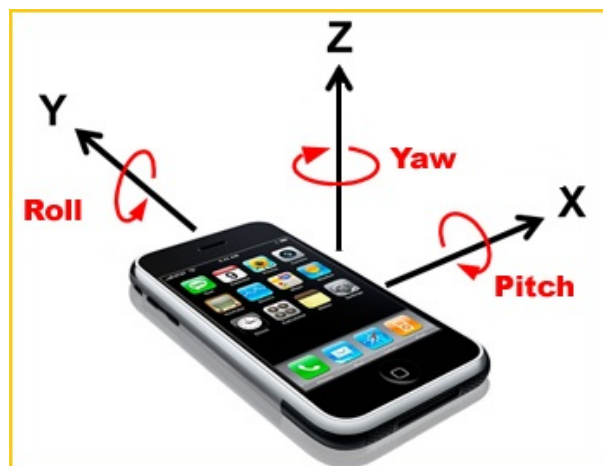
da conexão entre servidor e cliente estar estabelecida, o usuário pode manipular livremente a molécula com o celular de acordo com a tabela 6.1, basta manter o touchscreen do smartphone pressionado. Quando o usuário encontrar o átomo alvo, basta mover o volume de seleção na tela, o qual é controlado com o dispositivo de apontamento, apenas movendo o smartphone. No momento em que o átomo alvo estiver dentro do volume de seleção, basta realizar um duplo clique e ficar pressionando a tela do smartphone para iniciar a fase de desambiguação. Nessa fase, com os objetos dispostos no plano mapeado 1:1 para a tela do smartphone, o usuário só precisa deslizar seu polegar pelo touchscreen e acompanhar o cursor cilíndrico na monitor, quando o cursor colidir com o átomo alvo, basta levantar o dedo da tela do smartphone e a seleção será realizada.

Tabela 6.1 – Manipulação da Molécula

<i>Movimentos</i>	<i>Ações com mouse e menu moviment</i>	<i>Ações com Smartphone</i>
Up	Segurar e arrastar o mouse para cima/botão up	pitch para cima
Down	Segurar e arrastar o mouse para baixo/botão dow	pitch para baixo
Left	Segurar e arrastar o mouse para esquerda/botão left	yaw para esquerda
Right	Segurar e arrastar o mouse para direita/botão right	yaw para direita
Rot Left	botão rot left	roll sentido anti-horário
Rot Right	botão rot right	roll sentido horário

Fonte: O Autor

Figura 6.5 – Ilustração dos movimentos do smartphone



Fonte: Mobiforge

7 ESTUDO DE CASO - DOCAGEM MOLECULAR

Como dito na seção 1.1, docagem molecular é o método de prever a orientação preferencial de uma molécula à uma segunda quando vão conectar-se uma a outra para formar um conjunto estável. A docagem é um processo amplamente utilizado por químicos e bioquímicos para auxiliar na descoberta de novos fármacos. A docagem molecular também exerce um papel fundamental na transdução de sinal - processo no qual uma célula converte um tipo de sinal ou estímulo em outro - a orientação relativa das moléculas que estão por se ligar podem afetar o tipo de sinal produzido, e ainda, a docagem é útil para prever a intensidade e o tipo de sinal produzido. Em suma, todas essas simulações têm grandes contribuições no desenvolvimento de novos fármacos.

7.1 Algoritmo de Docagem

Para docagem molecular, existe um grande número de programas disponíveis, dentre os mais conhecidos livres para uso acadêmico podemos citar DOCK 6.0 (Kuntz, I.D., DOCK, UCSF Box 2240, UCSF, San Francisco, CA 94143-2240), Autodock4 (MORRIS et al., 2009), ZDOCK (CHEN; LI; WENG, 2003); e dentre os pagos GOLD (JONES et al., 1997), Dock-Vision (HART; NESS; READ, 1997), FlexE (CLAUSSEN et al., 2001). Diferentes programas usam diferentes algoritmos para realizar a docagem, entretanto a grande maioria está baseado em dois passos: algoritmo de busca e função de pontuação.

7.1.1 Algoritmo de Busca

O algoritmo de busca, na teoria, consiste em encontrar todas as possíveis orientações do ligante com o receptor, é importante ressaltar que ambos, ligante e receptor, são flexíveis e podem sofrer deformações. Por isso, com os recursos computacionais atuais, é praticamente inviável explorar exaustivamente todo o espaço de busca, pois como as moléculas podem sofrer deformações são muitas possibilidades e o tempo para testar todas é muito grande.

7.1.2 Função de Pontuação

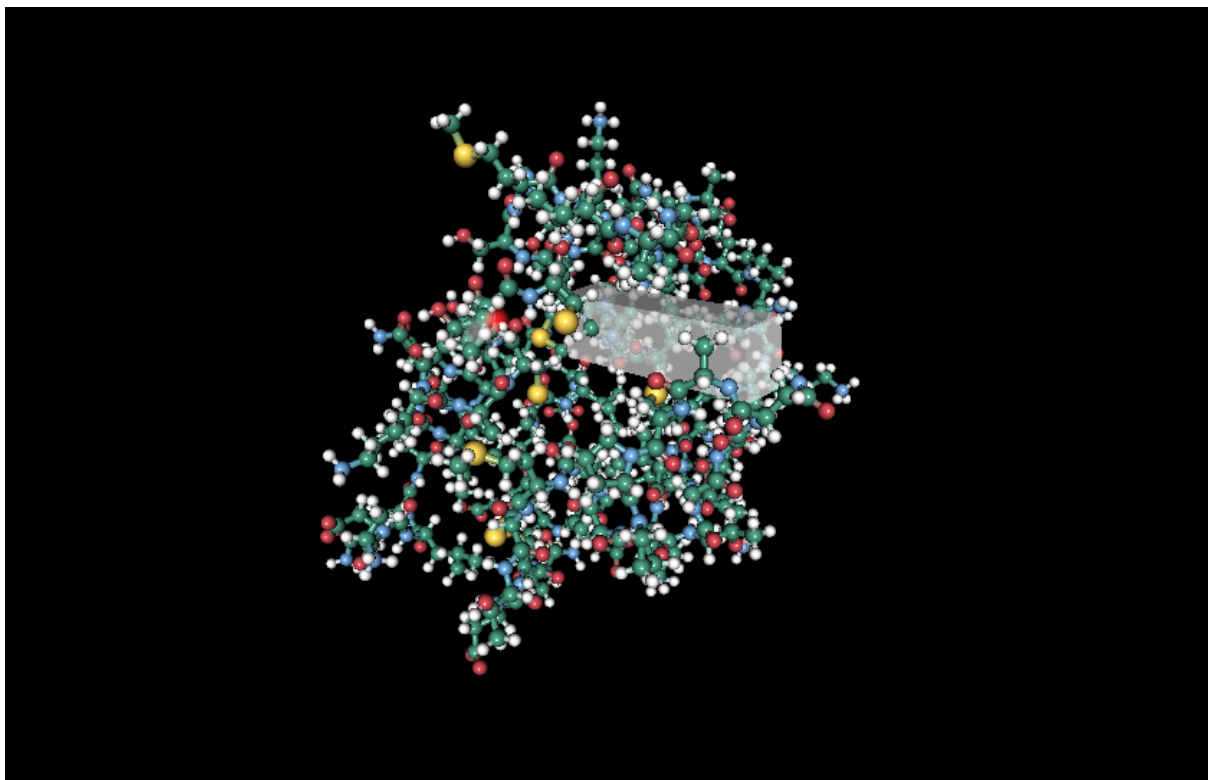
A função de pontuação recebe uma posição preferencial para ocorrer a ligação e retorna um número que indica a probabilidade dessa posição representar uma ligação favorável.

7.2 Busca Manual

Nosso trabalho propõe uma alternativa para o algoritmo de busca da docagem molecular, uma busca manual. Com o auxílio da técnica Dcanvas, um especialista com experiência no

processo, poderá encontrar a área favorável onde ocorrerá a docagem molecular. O especialista poderá manipular a molécula com o smartphone até visualizar a área desejada, depois com o auxílio do Dcavans seleciona os 8 pontos para delimitar essa área e a partir desses pontos nós plotamos o polígono que demarcar a área escolhida para se realizar a docagem molecular (figura 7.1).

Figura 7.1 – Exemplo de área selecionada para Docagem molecular.



Fonte: Printscreen Umol

7.3 Teste

Nossa proposta de teste é analisar manualmente uma molécula no visualizador Umol e, com o auxílio do Dcanvas, selecionar pontos para marcar a posição preferencial onde poderá ocorrer a docagem, plotar um polígono, obter o centro de massa e validar na ferramenta DockThor. DockThor é uma ferramenta online utilizada para realizar docagem molecular, nessa ferramenta é possível definir a área preferencial onde ocorrerá a docagem através de coordenadas X, Y e Z, que mais precisamente é o centro de massa de um polígono que delimita a área da docagem (GMMSB/LNCC,).

8 CONCLUSÕES E TRABALHOS FUTUROS

Nós apresentamos nesse trabalho a integração entre dois sistemas, Unitymol e Disambiguation Canvas. Mostramos como a técnica Dcanvas é adequada para realizar a interação e a seleção de átomos no visualizador de moléculas. Propusemos uma nova maneira de interação no Umol por meio de um smartphone. Estimamos que após essa integração podemos interagir com a molécula de maneira muito mais rápida e fácil. Também acrescentamos a possibilidade de seleção de átomos e com isso podemos usar a aplicação como uma ferramenta para facilitar algoritmos de docagem molecular.

Como trabalho futuro, pretendemos realizar testes com os especialistas na área molecular para, com isso, poder validar a nossa aplicação e medir seu potencial como uma real contribuição no processo de confecção de novos fármacos. Em relação a ferramenta, pretendemos também adicionar novos gestos para incluir novas funcionalidades, como por exemplo controlar o tamanho do volume de seleção através do *pinch* no touchscreen. Finalmente, pretendemos integrar o Dcanvas de forma oficial na Unitymol, a fim de disponibilizá-lo para a sua comunidade de usuários.

REFERÊNCIAS

- BERGMAN, L. D. et al. View: an exploratory molecular visualization system with user-definable interaction sequences. In: **ACM. Proceedings of the 20th annual conference on Computer graphics and interactive techniques**. [S.l.], 1993. p. 117–126.
- BERMAN, H. M. et al. The protein data bank. **Nucleic acids research**, Oxford Univ Press, v. 28, n. 1, p. 235–242, 2000.
- BOWMAN, D. A. et al. **3D user interfaces: theory and practice**. [S.l.]: Addison-Wesley, 2004.
- CHEN, R.; LI, L.; WENG, Z. Zdock: an initial-stage protein-docking algorithm. **Proteins: Structure, Function, and Bioinformatics**, Wiley Online Library, v. 52, n. 1, p. 80–87, 2003.
- CLAUSSEN, H. et al. Flexe: efficient molecular docking considering protein structure variations. **Journal of molecular biology**, Elsevier, v. 308, n. 2, p. 377–395, 2001.
- DEBARBA, H. G. Selection in 2d and 3d environments with levels of precision and progressive refinement. 2012.
- DELANO, W. L. The pymol molecular graphics system. 2002.
- GIORDAN, M. et al. Visualização de objetos moleculares: interfaces de ensino, pesquisa e extensão entre a educação em química ea engenharia de software. In: **Anais do Simpósio Brasileiro de Informática na Educação**. [S.l.: s.n.], 2004. v. 1, n. 1, p. 50–52.
- GMMSB/LNCC. Dockthor. Available from Internet: <<http://dockthor.lncc.br/>>.
- HART, T. N.; NESS, S. R.; READ, R. J. Critical evaluation of the research docking program for the casp2 challenge. **Proteins: Structure, Function, and Bioinformatics**, Wiley Online Library, v. 29, n. S1, p. 205–209, 1997.
- JONES, G. et al. Development and validation of a genetic algorithm for flexible docking. **Journal of molecular biology**, Elsevier, v. 267, n. 3, p. 727–748, 1997.
- LEE, S. J.; CHUNG, H. Y.; KIM, K. S. An easy-to-use three-dimensional molecular visualization and analysis program: Posmol. **BULLETIN-KOREAN CHEMICAL SOCIETY**, Korean Chemical Society, v. 25, n. 7, p. 1061–1064, 2004.
- LV, Z. et al. Game on, science-how video game technology may help biologists tackle visualization challenges. **PloS one**, Public Library of Science, v. 8, n. 3, p. e57990, 2013.
- MORRIS, G. M. et al. Autodock4 and autodocktools4: Automated docking with selective receptor flexibility. **Journal of computational chemistry**, Wiley Online Library, v. 30, n. 16, p. 2785–2791, 2009.
- SANDEEP, G. et al. Audocker le: A gui for virtual screening with autodock vina. **BMC research notes**, BioMed Central Ltd, v. 4, n. 1, p. 445, 2011.
- SEELIGER, D.; GROOT, B. L. de. Ligand docking and binding site analysis with pymol and autodock/vina. **Journal of computer-aided molecular design**, Springer, v. 24, n. 5, p. 417–422, 2010.

STEED, A. Towards a general model for selection in virtual environments. In: IEEE. **3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on**. [S.l.], 2006. p. 103–110.

TROTT, O.; OLSON, A. J. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. **Journal of computational chemistry**, Wiley Online Library, v. 31, n. 2, p. 455–461, 2010.