Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Ferramenta de Visualização Interativa de Comparação entre Múltiplos Genomas para a Identificação de Sintenias

Rodrigo Carneiro Munhoz Coimbra

Monografia apresentada como requisito parcial
para conclusão do Mestrado em Computação

Orientadora
Prof.ª Maria Emília M. T. Walter

Brasília
2010

Universidade de Brasília — UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Mestrado em Computação

Coordenador: Prof. Mauricio Ayala Rincón

Banca examinadora composta por:

Prof.ª Maria Emília M. T. Walter (Orientadora) — CIC/UnB
Prof. Nalvo Franco de Almeida Jr. — FACOM/UFMS
Prof. Marcelo de Macedo Brígido — CEL/UnB

CIP — Catalogação Internacional na Publicação

Coimbra, Rodrigo Carneiro Munhoz.

Ferramenta de Visualização Interativa de Comparação entre Múltiplos Genomas para a Identificação de Sintenias / Rodrigo Carneiro Munhoz Coimbra. Brasília : UnB, 2010.
111 p. : il. ; 29,5 cm.

Tese (Mestrado) — Universidade de Brasília, Brasília, 2010.

1. Bioinformática, 2. Visualização, 3. Sintenia, 4. Java

CDU 004

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro — Asa Norte
CEP 70910-900
Brasília–DF — Brasil

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Ferramenta de Visualização Interativa de Comparação entre Múltiplos Genomas para a Identificação de Sintenias

Rodrigo Carneiro Munhoz Coimbra

Monografia apresentada como requisito parcial
para conclusão do Mestrado em Computação

Prof.ª Maria Emília M. T. Walter (Orientadora)
CIC/UnB

Prof. Nalvo Franco de Almeida Jr.
FACOM/UFMS

Prof. Marcelo de Macedo Brígido
CEL/UnB

Prof. Mauricio Ayala Rincón
Coordenador do Mestrado em Computação

Brasília, 08 de junho de 2010

Resumo

A genômica comparativa é uma área de pesquisa que tem como objetivo a busca de como dados genômicos podem estar relacionados entre diferentes espécies. Particularmente, o volume de dados disponibilizados em bancos de dados públicos permite a busca de sintenias entre múltiplos genomas. Um par de genes é dito sintênico quando estes se conservam dentro da mesma região do DNA. Os métodos comparativos ainda usam abordagens tradicionais, tais como alinhamento textual e o uso de heurísticas para acelerar as comparações. Eles produzem como resultado arquivos textuais, o que dificulta análises mais específicas, tais como a identificação de sintenias. Nesse sentido, é essencial o desenvolvimento de ferramentas de visualização de comparações entre múltiplos genomas que facilitem a identificação de sintenias. O objetivo deste trabalho é propor e implementar uma ferramenta computacional que implemente um novo método de visualização que permite identificar sintenias entre múltiplos genomas, a partir de um genoma não totalmente sequenciado. Essa ferramenta foi aplicada na identificação de sintenias no fungo *P. brasiliensis*. Essa nova ferramenta, denominada Syntainia, está sendo desenvolvida como um software livre e já está disponível para download em `http://sourceforge.net/projects/syntainia`.

Palavras-chave: Bioinformática, Visualização, Sintenia, Java

Abstract

Comparative genomics is a research field that aims to find how genomic data can be related among different species. Particularly, the volume of data available in public databases allows for searching of synteny among multiple genomes. A pair of genes is said to be syntenic when they keep within the same region of the DNA. The comparative methods still use traditional approaches, such as text alignment and the use of heuristics to speed up comparisons. They produce results as text files, which makes difficult more specific analyses, such as the identification of synteny. Thus, it is essential the development of visualization tools to compare multiple genomes that facilitate the identification of synteny. The objective of this work is to propose and implement a software tool that implements a new visualization method for identifying synteny among multiple genomes from a not fully sequenced genome. This tool was applied in the identification of syntenies in the fungus *P. brasiliensis*. This new tool, called Syntainia, is being developed as free software and is now available for download at `http://sourceforge.net/projects/syntainia`.

Keywords: Bioinformatics, Visualization, Synteny, Java

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

A análise das informações geradas por projetos de sequenciamento de DNA constitui hoje um desafio sempre crescente para os cientistas da computação, uma vez que é essencial desenvolver software eficiente para processar o enorme volume de dados gerados por esses projetos (Mardis, 2008; McHardy, 2008; Morozova and Marra, 2008; Pop and Salzberg, 2007; Schuster, 2008). O processo de análise computacional de projetos de sequenciamento é basicamente dividido em duas fases. A fase de montagem tenta reconstruir grande pedaços das sequencias originais, enquanto a fase de anotação tem o objetivo de inferir funções biológicas e categorias para cada sequência montada, a fim de identificar genes, proteínas e RNAs não codificadores.

A fase de anotação é fortemente baseada em genômica comparativa. Em geral, a genômica comparativa tem como objetivo a busca de como dados genômicos (por exemplo, *loci* de nucleotídeos e genes, funções biológicas e categorias ontológicas) podem estar relacionados entre diferentes espécies. O principal objetivo é obter mais informação sobre uma espécie em estudo baseado no conhecimento prévio sobre espécies filogeneticamente relacionadas. As técnicas utilizadas em genômica comparativa precisam tratar enormes volumes de dados, no contexto das novas tecnologias de sequenciamento de alto desempenho.

Os métodos comparativos ainda usam abordagens tradicionais para comparar cadeias de nucleotídeos ou aminoácidos, executando alinhamento textual e utilizando heurísticas para acelerar as comparações. Por exemplo, as ferramentas BLAST (Altschul et al., 1997) e BLAT (Kent, 2002) produzem bons resultados, com uma saída bem detalhada. Porém, elas produzem como resultado apenas arquivos textuais, o que dificulta uma análise mais global, isto é, a compreensão de certas características que se revelam em grande porções de sequências de DNA. Particularmente, a ocorrência de sintenias não é fácil de se encontrar em arquivos textuais. Dois genes são sintênicos se eles estão localizados no mesmo cromossomo ou num de seus fragmentos, o que significa que eles devem estar na mesma cadeia de DNA (Passarge et al., 1999). Assim, uma ferramenta de visualização especialmente projetada pode suportar a descoberta de sintenias entre múltiplos genomas.

Diversas ferramentas de visualização foram desenvolvidas para a navegação em comparações entre dois genomas, através do alinhamento de suas sequências de nucleotídeos ou aminoácidos. Essa ferramentas apresentam gráficos extrema-

mente detalhados, contudo é difícil usá-los para o estudo sobre como genes estão organizados entre dois ou mais genomas diferentes ou para investigar se eles tem sua organização preservada. Fazem parte dessa categoria ferramentas como Apollo (Lewis et al., 2002), SyntenyView (Clamp et al., 2003), SyntenyVista (Hunt et al., 2004), ACT (Carver et al., 2005), SynBrowse (Pan et al., 2005), SynView (Wang et al., 2006), GBrowse_syn (McKay, 2007), Cinteny (Sinha and Meller, 2007), MEDEA (Broad Institute, 2009), Sybil (TIGR, 2009) e MizBee (Meyer et al., 2009) . Em geral, essas ferramentas apresentam muita informação que não é relevante no caso de estudo da conservação de genes entre diferentes espécies, Além disso, as ferramentas de visualização em geral são estáticas, não oferecendo mecanismo de interação com o usuário. Tais características motivam a busca por uma nova e mais clara forma de visualização da comparação entre múltiplos genomas.

A Universidade de Brasília mantém, em parceria com outras instituições, o projeto genoma do fungo *Paracoccidioides brasiliensis* (Felipe et al., 2005a). Mais recentemente, tem sido realizado um trabalho de genômica comparativa do genoma de *P. brasiliensis*, com o objetivo de identificar sintenias entre genomas de fungos não patogênicos e patogênicos humanos (de Carvalho, 2010). Um dos produtos desse trabalho foi a elaboração de uma técnica de visualização de como os genes de *P. brasiliensis* estão organizados dentro dos genomas de outros fungos, uma vez que o seu genoma não foi inteiramente sequenciado. Essa nova técnica é baseada no agrupamento de genes considerando seu posicionamento relativo nos cromossomos, *supercontigs* e *scaffolds*. Então, cada gene de um genoma é ligado com seu ortólogo em outro genoma. Essas linhas, formando caminhos entre os genomas, deixam claro como os genes são conservados entre diferentes espécies e a possibilidade de ocorrência de sintenias.

Nesse sentido, é relevante o desenvolvimento de uma ferramenta computacional que implemente o método de visualização elaborado por de Carvalho (2010) e o aperfeiçoe, de modo que seja capaz de apresentar a melhor visualização possível da comparação entre múltiplos genomas, de quaisquer organismos, de forma interativa e que apresente as informações no nível de detalhes que mais convier ao pesquisador.

Assim, o objetivo geral deste trabalho é apresentar um software de visualização de comparações entre múltiplos genomas, denominado Syntainia, que apresenta os genomas como grupos de fragmentos ligados a seus respectivos ortólogos, com o objetivo de facilitar a tarefa de identificação de sintenias.

São objetivos específicos deste projeto:

1. realizar um estudo comparativo entre as ferramentas de visualização mais comuns;
2. projetar a ferramenta com as seguintes características:
   (a) desenvolver uma ferramenta de visualização de fácil utilização, com uma interface gráfica intuitiva e integrada ao sistema operacional;
   (b) obter a melhor forma de visualizar os grupos de genes, utilizando algoritmos de baixa complexidade de tempo; e

   (c) prover capacidade de obtenção de informações sobre os genes a partir da visualização.

3. realizar um estudo de caso com o genoma do fungo *P. brasiliensis*; e
4. comparar as características da ferramenta implementada com outros softwares para visualização genômica.

O Capítulo 2 apresenta noções de Biologia Molecular necessárias à compreensão deste trabalho. Características e exemplos de ferramentas de visualização de comparações entre genomas são apresentados no Capítulo 3. Em seguida, no Capítulo 4, o Projeto Genoma Pb é apresentado, assim como os estudos mais recentes em genômica comparativa, que motivaram a elaboração deste trabalho. O Capítulo 5 apresenta o Syntainia, seu processo de desenvolvimento, requisitos, algoritmos, estruturas de dados e arquitetura. No Capítulo 6 é apresentado um estudo de caso sobre a adoção da ferramenta ao Projeto Genoma Pb. Finalmente, no Capítulo 7 são apresentadas as conclusões e delineados os trabalhos futuros.

Capítulo 2

Conceitos Básicos de Biologia Molecular

Este capítulo trata de conceitos básicos de Biologia Molecular, necessários à compreensão deste trabalho.

A Seção 2.1 apresenta as células e introduz os demais conceitos apresentados neste capítulo. Na Seção 2.2 as proteínas são apresentadas. Na Seção 2.3 são descritos os ácidos nucleicos (DNA e RNA), reservatórios moleculares da informação genética. Explanação acerca de genes, cromossomos e código genético é feita na Seção 2.4. O Dogma Central da Biologia Molecular é tratado na Seção 2.5. Por fim, considerações a respeito de técnicas de sequenciamento genético são feitas na Seção 2.6.

2.1 Células

A célula é a menor unidade que exibe o comportamento conhecido como vida (de Carvalho et al., 2004). Muitas moléculas encontradas no interior das células são macromoléculas, polímeros de alto peso molecular constituídos por precursores relativamente simples. As proteínas e os ácidos nucleicos, por exemplo, são formados pela polimerização de subunidades relativamente pequenas (Lehninger et al., 1995).

Embora os organismos vivos contenham um número muito grande de proteínas e de ácidos nucleicos, uma simplicidade fundamental está na base das suas estruturas. As subunidades monoméricas simples com as quais todas as proteínas e todos os ácidos nucleicos são construídos encontram-se em número pequeno e são idênticas em todas as espécies.

A sobrevivência de espécies biológicas requer que sua informação genética seja mantida em uma forma estável e, ao mesmo tempo, expressada com um número muito pequeno de erros. O armazenamento efetivo e a expressão acurada da mensagem genética definem cada espécie e sua continuidade por gerações sucessivas.

A partir dessas considerações, pode-se destacar alguns dos princípios da lógica molecular da vida (Lehninger et al., 1995):

1. Todos os organismos vivos têm os mesmos tipos de subunidades monoméricas.

2. A estrutura das várias macromoléculas biológicas revelam a existência de modelos subjacentes comuns.

3. A identidade de cada organismo é preservada pela posse de conjuntos característicos de ácidos nucleicos e proteínas.

2.2 Proteínas

As proteínas (do grego *protos*: a primeira; a mais importante) constituem, ao lado da água, a maior fração das células.

Quase tudo que ocorre nas células envolve uma ou mais proteínas. As proteínas têm muitas funções biológicas diferentes: algumas têm atividade catalítica e funcionam como enzimas; outras servem como elementos estruturais, como nutrientes e armazenamento; outras transportam sinais específicos ou substâncias específicas para o interior ou exterior das células, agem como defesa ou têm função reguladora.

O papel central ocupado por elas é evidenciado pelo fato de que a informação genética é, em última instância, expressa como proteína. Para cada proteína existe um segmento de DNA — um gene (Seção 2.4) — que guarda a informação, especificando sua sequência de aminoácidos. Em uma célula existem milhares de diferentes tipos de proteínas, cada uma delas codificada por um gene e, cada uma delas, executando uma função específica.

As proteínas são cadeias de aminoácidos. Cada aminoácido está unido a seus vizinhos por um tipo específico de ligação covalente (ligação peptídica).

Todas as proteínas, em todas as espécies, independente da função ou da atividade biológica, são constituídas com o mesmo conjunto de 20 aminoácidos (Figura 2.1).

Todos os 20 aminoácidos encontrados nas proteínas têm um grupo carboxila ($COOH$) e um grupo amina (NH_2) ligados ao mesmo átomo de carbono (o carbono $\alpha - C_\alpha$). Eles diferem uns dos outros por suas cadeias laterais ou grupos R, os quais variam em estrutura, tamanho e carga elétrica, e são responsáveis pelas diferentes características dos aminoácidos (tal como a solubilidade em água).

As proteínas diferem umas das outras porque têm um número e uma sequência de resíduos de aminoácidos — unidades de aminácidos que sofreram ligação peptídica, perdendo um átomo de hidrogênio de seu grupo amino e a parte hidroxila do seu grupo carboxila — que são diferentes entre si. Os aminoácidos são o alfabeto da estrutura proteica, podendo ser arranjados em um número quase infinito de sequências para fazer um número quase infinito de diferentes proteínas.

Conceitualmente, a estrutura das proteínas pode ser considerada em quatro níveis (Figura 2.2). A **estrutura primária** é definida pela sequência de aminoácidos unidos por ligações peptídicas; a **estrutura secundária** corresponde aos arranjos recorrentes no espaço de resíduos de aminoácidos adjacentes; a **estrutura terciária** é a organização tridimensional completa do polipeptídeo, referindo-se ao relacionamento espacial entre todos os seus aminoácidos; e, por fim, a **estrutura quaternária**, encontrada em proteínas com várias cadeias de polipeptídeos, é aquela que especifica a relação espacial dos polipeptídeos, ou subunidades, no interior da proteína (Lehninger et al., 1995).

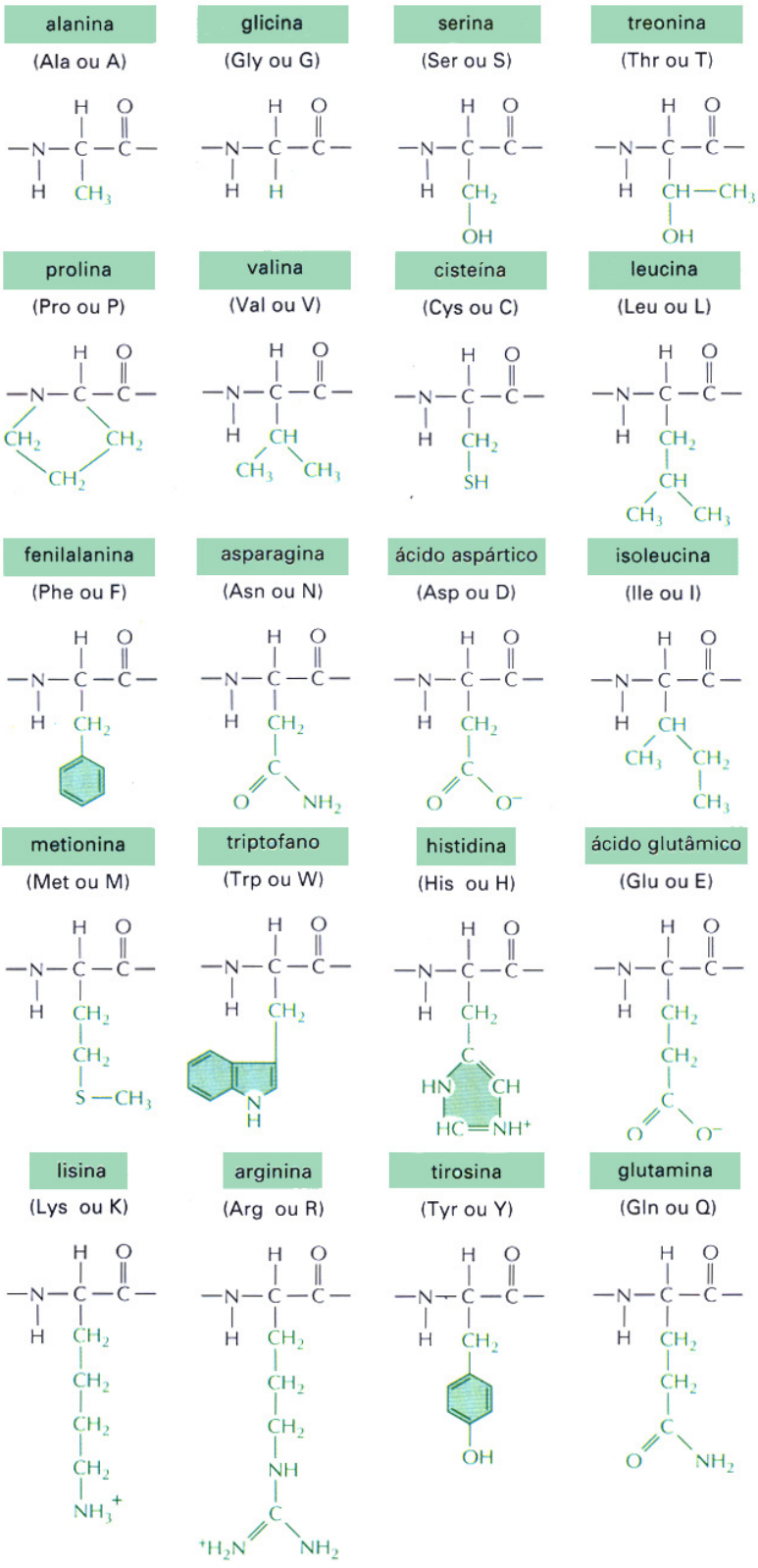


Figura 2.1: Os 20 aminoácidos das proteínas (abaixo do nome do aminoácido — e acima da estrutura molecular — são indicados seus códigos de três letras e de uma letra).

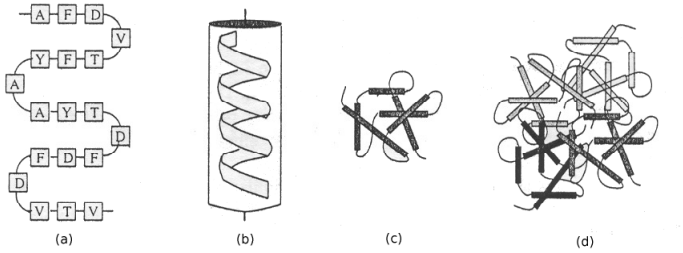


Figura 2.2: Estrutura das proteínas: (a) primária, (b) secundária, (c) terciária e (d) quaternária (Setubal and Meidanis, 1997).

Assim, um polipeptídeo com uma sequência específica de aminoácidos enovela-se em uma estrutura tridimensional única; e esta estrutura, por sua vez, determina a função da proteína.

A sequência de aminoácidos de uma proteína, ou sua *estrutura primária*, pode ser muito informativa para um biólogo. Nenhuma outra propriedade distingue tão claramente uma proteína de outra (Lehninger et al., 1995):

1. A estrutura tridimensional de uma proteína é determinada por sua sequência de aminoácidos.
2. A função de uma proteína depende de sua estrutura tridimensional.
3. A estrutura tridimensional de uma proteína é única, ou está muito próxima disso.
4. As forças mais importantes que estabilizam a estrutura tridimensional específica de uma dada proteína são as interações não-covalentes.
5. Finalmente, muito embora a estrutura das proteínas seja complicada, vários padrões característicos podem ser reconhecidos.

A sequência de aminoácidos em uma proteína pode fornecer pistas sobre a estrutura, a função, a localização celular e a evolução da proteína. A maior parte desses conhecimentos é obtida pela busca de similaridades com outras sequências conhecidas. Milhares de sequências são conhecidas e estão disponíveis em bancos de dados computadorizados (Boeckmann et al., 2003). A comparação de uma sequência recém-obtida com esse grande estoque de sequências geralmente revela relacionamentos que são tanto surpreendentes quanto esclarecedores.

A probabilidade de que uma dada informação a respeito de uma nova proteína possa ser deduzida da sua estrutura primária melhora constantemente com a adição quase diária de novas sequências de aminoácidos ao grande número daquelas já publicadas e armazenadas em bancos de dados públicos.

2.3 Ácidos Nucleicos

Os ácidos nucleicos, ácido desoxirribonucleico (DNA) e ácido ribonucleico (RNA), são polímeros de nucleotídeos. Eles são os reservatórios moleculares da informação

genética. A estrutura de toda proteína e, em última análise, de todo constituinte celular, é um produto da informação programada numa sequência nucleotídica dos ácidos nucleicos da célula.

Os nucleotídeos são compostos ricos em energia que direcionam os processos metabólicos (principalmente as biossínteses) em todas as células. Eles também funcionam como sinais químicos, elos importantes nos sistemas celulares que respondem a hormônios e outros estímulos extracelulares, além de serem componentes estruturais de vários cofatores enzimáticos e de intermediários metabólicos (Lehninger et al., 1995).

Cada nucleotídeo (Figura 2.3) é formado por três componentes:

1. uma base orgânica nitrogenada;
2. um açúcar de cinco átomos de carbono (pentose) numerados de 1' a 5'; e,
3. um grupo fosfato.

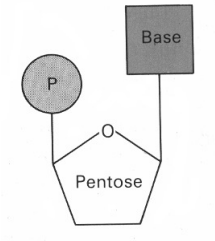


Figura 2.3: Estrutura de um nucleotídeo (Darnell et al., 1986).

As bases nitrogenadas são derivadas de dois compostos ancestrais, as *pirimidinas* e as *purinas*. Tanto o DNA quanto o RNA contêm duas bases púricas principais: a *adenina* (A) e a *guanina* (G). O DNA e o RNA possuem também duas pirimidinas principais; em ambos os tipos de ácidos nucleicos, uma delas é a *citosina* (C). A única diferença importante entre as bases do DNA e as do RNA é a natureza da segunda pirimidina: *timina* (T) no DNA e *uracila* (U) no RNA (Figura 2.4).

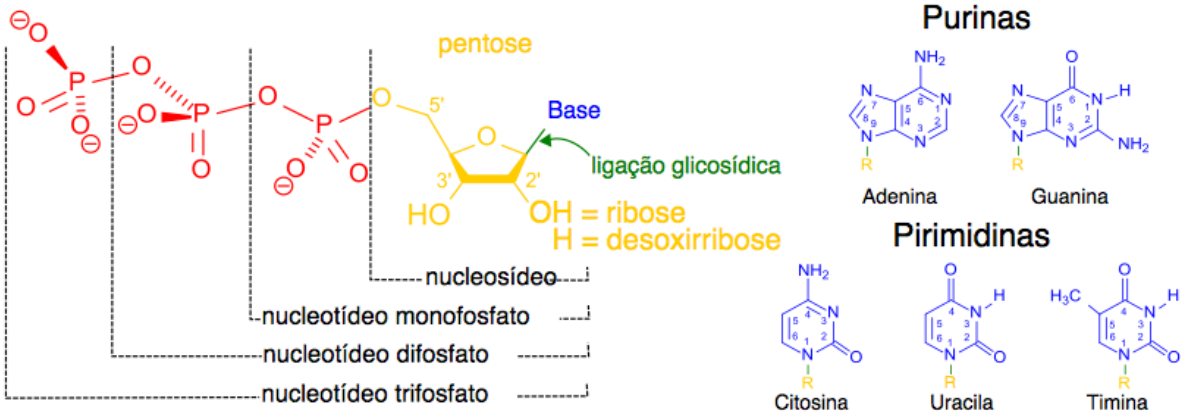


Figura 2.4: Elementos estruturais dos nucleotídeos mais comuns (Vickers (2007), com adaptações).

Duas espécies de pentose são encontradas nos ácidos nucleicos. O DNA possui 2'-Desoxi-D-Ribose (desoxirribose) e o RNA contém D-Ribose (ribose) (Figuras 2.4 e 2.5).

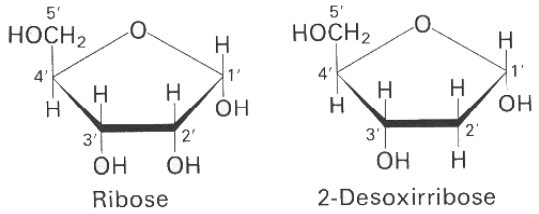


Figura 2.5: Açúcares presentes nos ácidos nucleicos (Setubal and Meidanis, 1997).

Os nucleotídeos sucessivos, tanto no DNA quanto no RNA, são ligados através de *pontes* de grupos fosfato. O grupo hidroxila (OH) do carbono 3' de um nucleotídeo se liga ao grupo fosfato do carbono 5' (*ligação fosfodiéster*). Desta forma, o esqueleto covalente dos ácidos nucleicos consiste de resíduos fosfato e pentose alternantes e as bases características podem ser consideradas como grupos laterais unidos ao esqueleto a intervalos regulares.

Todas a ligações fosfodiésteres nas fitas do DNA e RNA possuem a mesma orientação ao longo da cadeia, conferindo a cada fita linear do ácido nucleico uma polaridade e extremidades 5' e 3' distintas. Portanto, uma cadeia de ácidos nucleicos tem uma orientação química, que por convenção começa no carbono 5' livre e acaba na terminação que contém o carbono 3' livre.

2.3.1 DNA

Como repositório da informação genética, o DNA ocupa uma posição única e central entre as macromoléculas biológicas. As sequências nucleotídicas do DNA descrevem as estruturas primárias de todos os RNA e proteínas celulares, e através das enzimas é capaz de controlar o tipo e a quantidade de todos os componentes celulares, determinando em última instância as características fenotípicas todo ser vivo (Lehninger et al., 1995).

Dessa forma, é na molécula de DNA que estão codificadas as estruturas das proteínas, geradas a partir da transcrição de DNA em RNA e da tradução deste em proteínas (Setubal and Meidanis, 1997), o chamado *Dogma Central da Biologia Molecular* (Seção 2.5).

O armazenamento da informação biológica é a única função conhecida do DNA.

O modelo tridimensional para a estrutura do DNA consiste de duas cadeias helicoidais que se enrolam ao redor do mesmo eixo, formando uma dupla hélice que gira no sentido da mão direita (Figura 2.6). As bases púricas e pirimídicas de ambas as fitas estão empilhadas dentro da dupla hélice, com suas estruturas muito próximas e perpendiculares ao longo do eixo da hélice. As duas cadeias ou fitas da hélice são *antiparalelas*, ou seja, suas ligações 5',3' correm em direções opostas. As fitas são complementares entre si: toda vez que aparecer uma adenina numa cadeia, timina será encontrada na outra, onde se encontrar guanina numa cadeia,

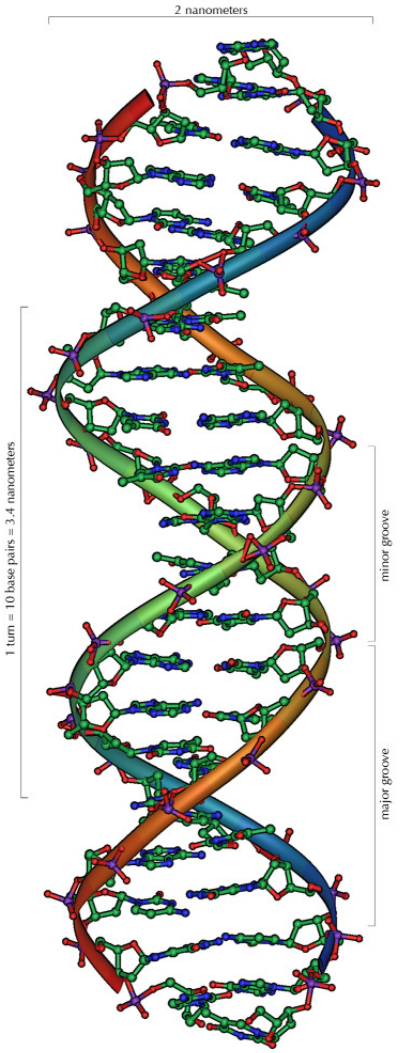


Figura 2.6: Uma visão geral da estrutura do DNA (Ströck, 2006).

encontrar-se-á citosina na outra. O DNA pode ser pensado como uma cadeia linear de letras (de Carvalho et al., 2004), como no exemplo da Figura 2.7.

$$5' \ldots \text{GGATCAGCGC} \ldots 3'$$
$$3' \ldots \text{CCTAGTCGCG} \ldots 5'$$

Figura 2.7: As duas fitas do DNA são uma o complemento reverso da outra (com relação às regras de emparelhamento de bases: A=T e C≡G).

2.3.2 RNA

O RNA é uma molécula que tem estrutura semelhante à do DNA (Figura 2.8), com algumas diferenças de composição, quanto à estrutura. O RNA normalmente é formado por uma fita simples de nucleotídeos.

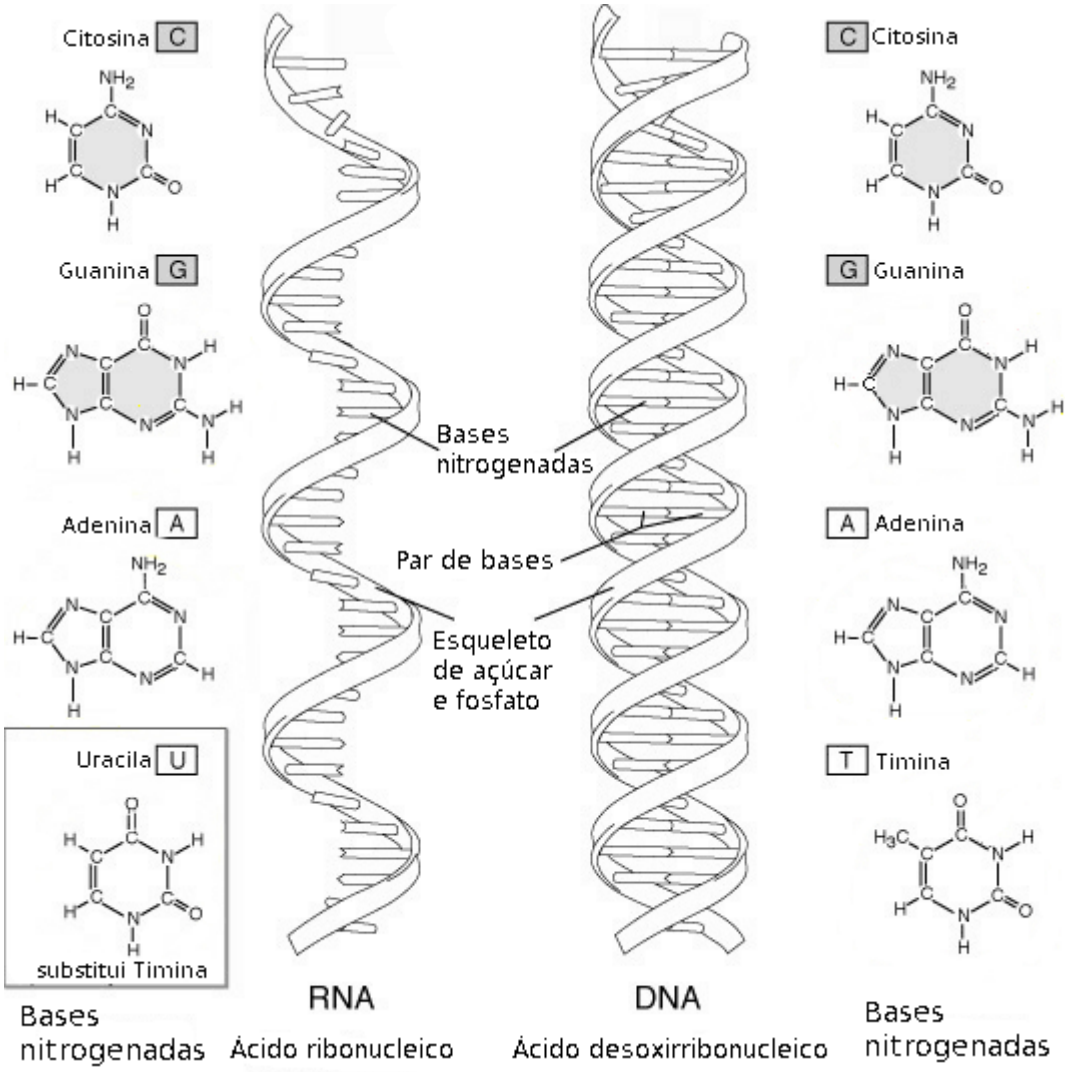


Figura 2.8: RNA e suas bases nitrogenadas à esquerda e DNA à direita (Access Excellence (2010), com adaptações).

Várias classes de RNAs são encontradas na célula, cada uma com uma função distinta. Os *RNAs ribossômicos* (rRNA) são componentes estruturais dos ribossomos, grandes complexos que realizam a síntese de proteínas. Os *RNAs mensageiros* (mRNA) são ácidos nucleicos que transportam a informação de um ou de uns poucos genes até o ribossomo, onde as proteínas correspondentes serão sintetizadas. Os *RNAs transportadores* (tRNA) são moléculas adaptadoras que traduzem a informação presente no mRNA numa sequência específica de aminoácidos.

2.4 Genes, Cromossomos e Código Genético

Como dito anteriormente, a informação sequencial necessária à formação de proteínas ou de RNA é encontrada nas sequências nucleotídicas correspondentes no

DNA. Um segmento de DNA que contém a informação necessária para a síntese de um produto biológico funcional (proteína ou RNA) é referido como *gene*.

Diz-se que um gene que está dando origem a um produto biológico está sendo *expresso*. Células diferentes e em estágios de desenvolvimento ou condições diferentes expressam genes distintos e em intensidades diversas.

Uma célula típica possui muitos milhares de genes e as moléculas de DNA, nada surpreendentemente, tendem a ser muito longas (Lehninger et al., 1995). Estima-se que o genoma humano tenha em torno de 30.000 genes (International Human Genome Sequencing Consortium, 2001).

Nos organismos eucariontes (cuja(s) célula(s) possuem núcleo delimitado), os genes são compostos de partes chamadas *íntrons* e *éxons*, que se alternam dentro do gene. Na transcrição (Seção 2.5), os íntrons são retirados do mRNA (*splicing*). Assim, os íntrons correspondem a porções que não são utilizadas na síntese da proteína codificada pelo gene e os éxons correspondem à porção do DNA que originará proteínas. A fração do DNA que corresponde a um gene completo é chamada *DNA genômico*, já uma porção que corresponde ao gene sem os íntrons é chamada de *DNA complementar* (cDNA). O cDNA pode ser obtido a partir do mRNA através do processo chamado *transcrição reversa*.

As moléculas de DNA são usualmente "empacotadas" em estruturas chamadas de *cromossomos*. A maioria das bactérias e vírus possuem um único cromossomo; os eucariontes usualmente possuem muitos. Um único cromossomo tipicamente contém milhares de genes individuais. O conjunto completo de cromossomos de uma célula — incluindo todos os genes e DNA intergênicos (que está entre os genes) em todos os cromossomos de uma célula — é referido como *genoma* celular.

As proteínas são sintetizadas com uma sequência particular de aminoácidos, através da *tradução* da informação codificada no RNA mensageiro. Para especificar uma proteína, basta especificar os aminoácidos que ela contém. Os aminoácidos são especificados por unidades informacionais no mRNA chamadas de *códons*. Os códons para os aminoácidos consistem de trincas nucleotídicas específicas. A tradução requer moléculas adaptadoras, os RNAs transportadores, que reconhecem códons e inserem aminoácidos em suas posições sequenciais apropriadas no polipeptídeo. A tabela que relaciona os códons aos aminoácidos é chamada de *código genético* (Tabela 2.1). O código genético é degenerado, significando que um aminoácido pode ser especificado por mais de um códon (neste caso, degenerado não significa imperfeito, nem ambíguo, porque nenhum códon especifica mais de um aminoácido).

As palavras do código genético padrão são provavelmente universais em todas as espécies, embora alguns desvios menores existam na mitocôndria e em uns poucos organismos unicelulares.

Na síntese de proteínas, uma *fase de leitura* é uma das três possíveis formas de agrupar as bases para formar códons em uma sequência de DNA ou RNA. Considerando, por exemplo, a sequência GGATCAGCGC da Figura 2.7. Uma possível fase de leitura seria GGA, TCA, GCG, ignorando a última base C, em que foram formados códons a partir da primeira base; outra fase de leitura seria feita ignorando-se a primeira base G e agrupando as demais bases nos seguintes códons GAT, CAG, CGC. Uma terceira fase possível ignoraria as duas bases GG formando ATC, AGC,

Tabela 2.1: Tabela do código genético de códons mapeados em aminoácidos (Setubal and Meidanis, 1997).

Primeira Posição	*Segunda Posição*				*Terceira Posição*
	G	A	C	U	
G	Gly	Glu	Ala	Val	G
	Gly	Glu	Ala	Val	A
	Gly	Asp	Ala	Val	C
	Gly	Asp	Ala	Val	U
A	Arg	Lys	Thr	Met	G
	Arg	Lys	Thr	Ile	A
	Ser	Asn	Thr	Ile	C
	Ser	Asn	Thr	Ile	U
C	Arg	Gln	Pro	Leu	G
	Arg	Gln	Pro	Leu	A
	Arg	His	Pro	Leu	C
	Arg	His	Pro	Leu	U
U	Trp	STOP	Ser	Leu	G
	STOP	STOP	Ser	Leu	A
	Cys	Tyr	Ser	Phe	C
	Cys	Tyr	Ser	Phe	U

e desprezando as duas bases finais GC. Dessa forma, existem três possíveis fases de leitura da sequência de bases do DNA, iniciando na primeira, segunda ou terceira letras da sequência. A partir da quarta letra as fases de leituras são iguais a uma das três primeiras fases, com um ou mais códons a menos.

Levando-se em consideração a fita complementar de uma sequência de DNA, as fases de leitura devem ser consideradas, também no sentido reverso. Assim, tem-se mais três fases, num total de seis possíveis fases de leitura.

Uma *Open Reading Frame* – ORF, ou *fase aberta de leitura*, é uma sequência que começa no códon inicial de um gene com comprimento múltiplo de três, sendo completamente mapeada em códons, sem precisar ignorar nenhuma base no final da sequência (Setubal and Meidanis, 1997).

2.5 Transcrição, Tradução e Síntese Proteica – O Dogma Central da Biologia Molecular

O conhecimento da estrutura do DNA levou às questões sobre a sua função. A própria estrutura do DNA sugeriu como ele poderia ser copiado, de forma que a informação nele contida pudesse ser transmitida de uma geração para a seguinte. Compreender como a informação no DNA era convertida em proteínas funcionais tornou-se possível através da descoberta do mRNA, do tRNA e a solução do código genético.

Estes e outros avanços importantes levaram ao *Dogma Central da Biologia Molecular*, que define três processos principais na utilização celular da informação genética. O primeiro é a *replicação*, processo de cópia do DNA pai para formar as moléculas filhas de DNA, tendo sequências nucleotídicas idênticas. O segundo é a *transcrição*, processo pelo qual partes da mensagem genética codificada no DNA são copiadas precisamente, na forma de RNA. O terceiro é a *tradução*, na qual a mensagem genética codificada no mRNA é traduzida, nos ribossomos, numa proteína com uma sequência específica de aminoácidos.

2.5.1 RNAs não codificadores

Além do importante papel na *tradução*, existe uma classe de moléculas de RNA que não é traduzida em proteína, o chamado RNA não codificador (ncRNA). A classe de moléculas de ncRNAs pode ser dividida em dois grandes grupos: manutenção (*housekeeping*) e regulação. Os ncRNAs de manutenção incluem todas as classes de RNAs envolvidos no processo de transcrição primária, tradução e controle de qualidade de traduções. Os ncRNAs de regulação constituem um grupo muito mais diversificado, que compreende os ncRNAs envolvidos em uma regulação específica de vários aspectos dos genes expressos, tanto nos procariotos quanto nos eucariotos. Os níveis em que os RNAs reguladores podem influenciar processos celulares variam da regulação da transcrição ao controle da tradução. O estudo de como as moléculas de ncRNA atuam nas células é uma área de estudo em foco na atualidade, devido ao papel cada vez mais claro que essas moléculas desempenham no Dogma Central da Biologia Molecular (Mattick, 2003).

2.6 Sequenciamento de Genomas

Na sua capacidade de reservatório da informação, a mais importante propriedade de uma molécula de DNA é a sua sequência nucleotídica. Até o final dos anos de 1970, obter-se a sequência de um ácido nucleico contendo mesmo 5 ou 10 nucleotídeos era difícil e muito laborioso. O desenvolvimento de novas técnicas tornou possível sequenciar moléculas de DNA cada vez maiores, com uma facilidade não imaginada algumas décadas antes. As técnicas dependeram de uma melhora na compreensão da química dos nucleotídeos, do metabolismo do DNA e em métodos que permitiram a separação das fitas do DNA.

Desde a década de 1990, técnicas em laboratório tornaram possível extrair o DNA ou o RNA de células, separar as duas fitas que formam o DNA, induzir a união de fitas simples de DNA que tenham sequências complementares de bases, cortar o DNA em pontos específicos ou aleatórios, copiá-lo, estimar seu tamanho e marcá-lo com isótopos radioativos ou corantes fosforescentes que permitem posterior detecção, sintetizar pequenas cadeias de DNA com a sequência de bases que se desejar, separar moléculas de DNA em função do seu tamanho aproximado e sequenciar o DNA, isto é, obter a sequência de bases que o compõem (de Carvalho et al., 2004).

Entretanto, mesmo com todos esses avanços, o primeiro problema que surge no processo de sequenciamento está justamente no processo experimental necessário para se extrair a sequência de bases nucleotídicas do DNA. Limitações técnicas impedem que se sequencie regiões com tamanhos maiores do que 1.000 bases por vez. Desta forma, para viabilizar o sequenciamento completo do genoma é necessário, primeiramente, tratar o DNA das células de forma a criar inúmeros fragmentos, que devem ser individualmente sequenciados e, posteriormente, montados como verdadeiras peças de um quebra-cabeças (Pappas Jr., 2003).

2.6.1 Sequenciamento Sanger

O primeiro organismo a ter seu DNA sequenciado foi o vírus Φ-X174 pelo pesquisador Frederick Sanger em 1975 (Sanger et al., 1977). O método desenvolvido por Sanger foi aperfeiçoado, com a utilização de sequenciadores automáticos e sistemas computacionais, e mais tarde utilizado no sequenciamento do genoma humano (International Human Genome Sequencing Consortium, 2001; Venter et al., 2001) e de inúmeros outros seres, inclusive o fungo *P. brasiliensis* (Felipe et al., 2003).

A seguir, o método Sanger é descrito, tanto suas etapas laboratoriais, quanto as etapas que demandam a utilização da sistemas computacionais.

Quebra da molécula de DNA

Para a quebra do DNA empregam-se técnicas tais como o uso de *endonucleases de restrição* e o método de *shotgun*.

As endonucleases de restrição clivam o DNA em sequências específicas para gerar um conjunto de fragmentos menores. Já no método de *shotgun*, uma solução contendo DNA purificado é submetido a algum procedimento que induza a quebra desordenada das moléculas (como, por exemplo, uma alta frequência de oscilação/vibração), que são posteriormente filtradas e separadas para processamento.

Replicação de DNA

Para a realização dos experimentos laboratoriais com DNA é necessária uma porção mínima de material. é preciso, também, que se tenha material disponível para repetição do experimento. Para tanto, utilizam-se técnicas, como a do *DNA recombinante* e da *Reação em Cadeia de Polimerase* (PCR), para clonagem do DNA.

A primeira etapa na clonagem de um gene é frequentemente a construção de uma biblioteca de DNA que inclua fragmentos representando a maioria do genoma de uma dada espécie. A biblioteca pode ser limitada a exprimir genes pela clonagem de apenas cópias do DNA complementar a mRNAs, isolados para construir uma biblioteca de cDNA. Um segmento específico de DNA pode ser amplificado e clonado usando a PCR.

Vetores de expressão fornecem as sequências requeridas para a transcrição, tradução e regulação dos genes clonados. Eles permitem a produção de grandes quantidades de proteínas clonadas.

A clonagem pela técnica do *DNA recombinante* envolve a separação de um gene específico ou segmento de DNA do seu cromossomo maior, a sua ligação a uma molécula de DNA transportadora pequena e depois a replicação deste DNA modificado, milhares ou mesmo milhões de vezes. O resultado é uma amplificação seletiva de um gene ou segmento de DNA particular. Esse tipo de clonagem acarreta cinco procedimentos gerais:

1. Um método para cortar o DNA em localizações precisas (com o uso de endonucleases de restrição).
2. Um método para unir dois fragmentos de DNA (o que é feito pela DNA ligase).
3. A seleção de uma pequena molécula de DNA capaz de auto-replicação. Segmentos de DNA a serem clonados podem se unir a DNA de vetores (plasmídeos, vírus). As moléculas de DNA compostas são chamadas de *DNA recombinante*.
4. Um método para realizar a transferência do DNA recombinante para a célula hospedeira, que fornecerá a "maquinaria" enzimática para a replicação do DNA.
5. Métodos para selecionar as células hospedeiras que contenham o DNA recombinante.

O princípio da técnica conhecida como PCR (*Polymerase Chain Reaction*) é baseado na estrutura e na sequência do DNA. Um conjunto de *primers* ou *iniciadores*, compostos por duas pequenas sequências de oligonucleotídeos complementares a uma certa extensão em ambos os lados do DNA a ser amplificado, é usado para iniciar a reação. O DNA a ser copiado é chamado de *DNA template* ou *molde*. A molécula de DNA é aquecida até que a hélice se desfaça e haja separação da fitas (*desnaturação do DNA* ou *melting*). A solução é vagarosamente resfriada, cada hélice encontra-se com a sua complementar e a estrutura de dupla hélice reconstitui-se (*anelamento*), permitindo a hibridização entre a hélice que é unitária e o *primer*. Esta preferência pelo *primer* à cromátide irmã ocorre devido à alta concentração dos *primers* no meio. Nucleotídeos livres em alta concentração são disponibilizados no meio e servirão na composição da nova sequência a ser replicada, no processo de *extensão da cadeia*.

O conjunto de reações em série de desnaturação, anelamento e extensão é definido como *um ciclo*. A execução de um ciclo resulta na amplificação da sequência de DNA desejada.

O produto de um ciclo de ampliação serve como molde para o próximo. Assim, a cada ciclo sucessivo dobra a quantidade de DNA.

Sequenciamento automatizado

O sequenciamento do DNA está automatizado desde a década de 1990, com o desenvolvimento do Projeto Genoma Humano (International Human Genome Sequencing Consortium, 2001; Venter et al., 2001). Esta tecnologia permite que a sequência de milhares de nucleotídeos possa ser obtida em algumas horas e projetos de sequenciamento muito grandes possam ser contemplados (Figura 2.9).

Figura 2.9: Sequenciador automático de DNA MegaBACE 1000 (General Electric Company, 2010).

Sequenciadores automáticos geram um arquivo compactado contendo *eletroferogramas* de uma placa, que por sua vez contém vários fragmentos de DNA. Um eletroferograma é composto de quatro gráficos coloridos, cada um corresponde a uma das quatro bases, A (Adenina), C (Citosina), G (Guanina) e T (Timina) (Figura 2.10). Quando uma base é identificada em uma posição do fragmento, o gráfico apresenta um pico na posição correspondente. Se uma base particular não pode ser identificada, o caracter N (uNknow) é associado à posição correspondente. Assim, estes eletroferogramas, armazenados em arquivos compactados, são enviados por meio eletrônico aos laboratórios de Bioinformática para dar início ao processo de análise computacional das sequências.

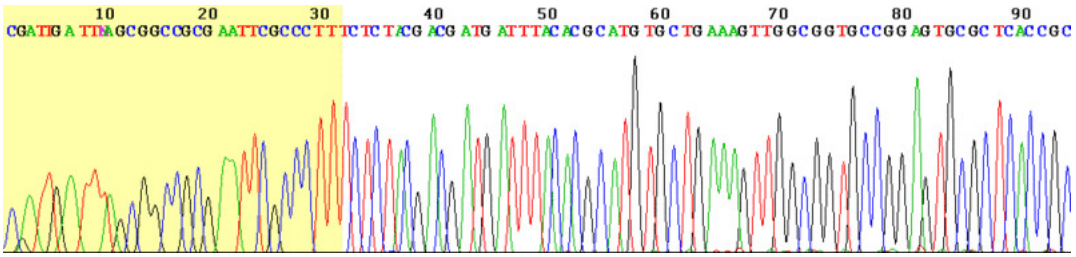


Figura 2.10: Exemplo de um eletroferograma (Loris, 2005).

Pipeline de Sequenciamento

Um *pipeline*, às vezes referenciado como *workflow* (Lemos, 2004), corresponde a uma sequência de processamento, na qual o resultado (saída) de uma etapa serve como entrada para outra etapa (Coimbra et al., 2007).

Tipicamente, um sistema de Bioinformática processa os fragmentos de DNA em três fases: submissão, montagem e anotação — que correspondem ao *pipeline* de sequenciamento, sendo que cada uma das fases é, por si só, formada por seu próprio *pipeline*. Estas fases tem por objetivo produzir sequências de caracteres correspondentes aos fragmentos gerados nos laboratórios de Biologia Molecular, recompor trechos do DNA original e identificar funções e categorias nestes trechos de sequências identificados. A seguir são descritas cada uma das etapas do *pipeline*, bem como citadas ferramentas computacionais utilizadas nas mesmas.

Submissão

Na fase de submissão, cada eletroferograma de uma placa é descompactado e transformado em uma cadeia chamada *read*. Em uma *read*, para cada base da sequência é associado um valor referente à probabilidade de erro na identificação da base nitrogenada identificada. As expressões *sequência* e *read* serão utilizadas como sinônimos. Normalmente, a tarefa de converter o arquivo binário que representa uma *read* para um formato legível por pessoas é feita pelo programa Phred (Ewing and Green, 1998; Ewing et al., 1998), que gera para cada *read* um arquivo no formato *phd*, que contém uma cadeia composta pelos caracteres A, C, G, T e N e a probabilidade de erro associada a cada base. O programa Phd2Fasta (Green, 2006) converte os arquivos *phd* em arquivos tipo texto no formato *FASTA* (NCBI, 2006), gerando um par de arquivos para cada conjunto de arquivos *phd* de uma placa: um arquivo contendo a cadeia de caracteres (arquivo de sequências) e outro com as respectivas probabilidades de erro (arquivo de qualidade).

Cada sequência é filtrada para remover porções que provavelmente não pertencem ao organismo sendo estudado, mas que pertencem a vetores (sequências dos organismos usados para replicar o DNA do organismo que está sendo estudado) e contaminantes (sequências de DNA de outros organismos). As *reads* são filtradas utilizando-se programas tais como o Cross_match (Green, 2006). Os trechos identificados como sendo de outros organismos são mascarados com o caractere X.

Em alguns casos, pode ser feita uma análise de redundância entre as sequências de uma placa. Para tanto, pode-se executar um programa de montagem de sequências, como o CAP3 (Huang and Madan, 1999). Os agrupamentos gerados pelo CAP3 indicam sequências muito similares e provavelmente redundantes dentro da placa submetida.

Finalmente, as sequências são armazenados em um banco de dados juntamente com estatísticas sobre a placa (por exemplo, o total de *reads*). Outras informações, como por exemplo a redundância, também podem ser armazenadas, de acordo com as necessidades de cada projeto de sequenciamento. A Figura 2.11 ilustra uma fase de submissão em que é feita análise de redundância.

Montagem

A fase de montagem consiste em gerar agrupamentos de sequências similares, isto é, sequências que têm prefixos e sufixos "aproximadamente iguais". Duas sequências são semelhantes se há similaridades entre o sufixo de uma e o prefixo de outra. Esta correspondência é conhecida como *alinhamento*. Estes agrupamentos buscam

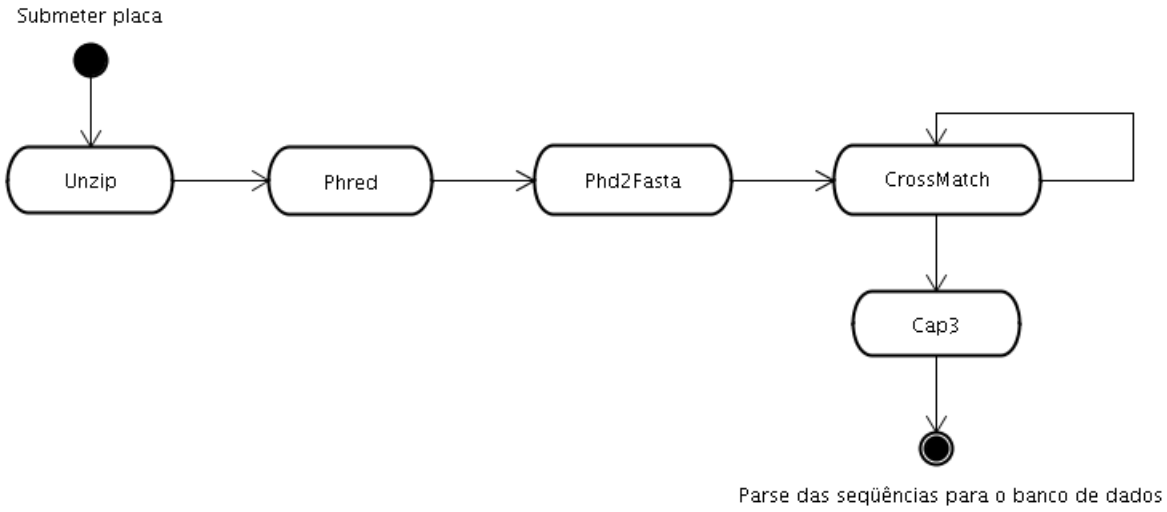


Figura 2.11: Exemplo de sequência de programas executados na fase de submissão.

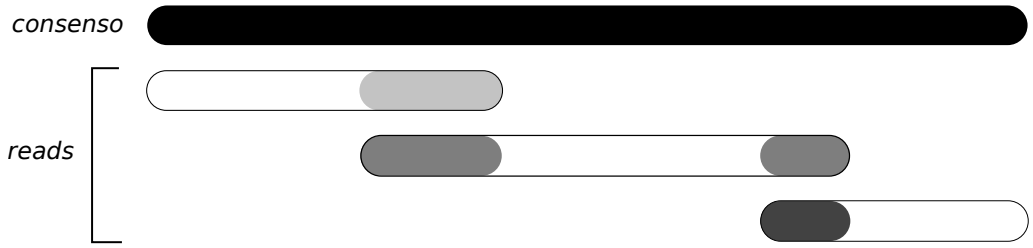


Figura 2.12: Alinhamento de um *contig*, resultante do processo de agrupamento de sequências, que baseia-se na similaridade entre prefixos e sufixos das sequências.

unir fragmentos que "potencialmente" pertencem à mesma região do DNA. Grupos formados por mais de uma sequência são chamados *contigs* (Figura 2.12) e grupos formados por uma única sequência são chamados *singlets*. Para cada *contig*, uma sequência *consenso* é gerada e esta representa o *contig*.

Os programas Phrap (Green, 2006) e CAP3 (Huang and Madan, 1999) são normalmente usados para "montar" as sequências. Ambos geram um arquivo com o formato *ace*, contendo dados sobre a montagem e os alinhamentos das sequências que compõem um *contig*, e outro arquivo que contém as sequências dos *singlets* em formato *FASTA*. Outros arquivos também são gerados, porém apenas aqueles dois são utilizados nos processamentos seguintes.

A identificação de possíveis genes nas sequências (sequências *consenso* ou *singlets*) é feita através de programas como o Glimmer (Delcher et al., 1999; Salzberg et al., 1998), que identifica posições iniciais ou finais de uma região que possivelmente esteja codificando um gene. As posições de cada "candidato" a gene são armazenadas em um banco de dados, juntamente com as sequências *consenso* e os *singlets*. Nos projetos de ESTs não é necessário identificar genes, uma vez que as ESTs representam porções do DNA que são expressas (já correspondem, portanto, a genes), assim, neste caso, apenas as sequências *consenso* e os *singlets* são

armazenados no banco de dados.

Ao término da fase de montagem, são também armazenadas algumas estatísticas, como o número total de grupos (*contigs* e *singlets*) e o número total de genes identificados (no caso de DNA genômico). Algumas visualizações de *contigs* também são podem ser disponibilizadas, mostrando o alinhamento das *reads* para a formação da sequência *consenso*.

Anotação

O objetivo da fase de anotação é identificar funções e categorias das sequências geradas na fase de montagem. É geralmente dividida em dois passos. Primeiro, a anotação automática, em que devem ser comparadas todas as sequências do projeto com sequências de bancos de dados públicos. As funções e categorias das sequências estudadas são inferidas por comparações com sequências semelhantes que tiveram suas funções e categorias previamente determinadas. O segundo passo, a anotação manual, é feita pelos biólogos, que utilizam as informações da anotação automática, bem como seus conhecimentos, para inferir a função associada à sequência. Estas informações também são armazenadas no banco de dados do projeto.

A anotação automática utiliza programas como o BLAST (Altschul et al., 1990) e o FASTA (Pearson and Lipman, 1988). As sequências encontradas nos bancos de dados públicos com maior semelhança às sequências estudadas identificadas pelo BLAST ou pelo FASTA são chamadas *best hits*. Estes programas fornecem saídas no formato HTML, que podem ser armazenados diretamente no banco de dados; em formato texto simples, que podem ser depois processadas para melhor visualização, ou até mesmo em formato XML (este último apenas pelo BLAST).

2.6.2 Sequenciamento de Alto Desempenho

O sequenciamento Sanger tem sido o método mais utilizado pelos pesquisadores nos últimos anos. Contudo, novos métodos de sequenciamento tem sido desenvolvidos. Esses novas tecnologias rapidamente ganharam espaço entre os pesquisadores devido à capacidade de sequenciamento de milhões de sequências a um custo muito baixo, em comparação ao método Sanger. Esses novos métodos tem tido um grande impacto nas áreas de pesquisa relacionadas a sequenciamento de DNA, abrindo novas frentes de pesquisa, tais como o estudo de DNAs conservados de espécies já extintas, como o mamute, e a caracterização da diversidade ecológica por meio do sequenciamento de DNA de amostras ambientais (Alvarez, 2009; Mardis, 2008).

Um dos equipamentos que implementa um dos novos métodos de sequenciamento é o Roche/454 FLX (Figura 2.13). Este sequenciador foi introduzido em 2004, e utiliza uma técnica de sequenciamento conhecida como pirosequenciamento. No pirosequenciamento a incorporação de cada nucleotídeo a uma fita de DNA, por meio da enzima DNA polimerase, acarreta a liberação de pirofosfato. Esta molécula, por sua vez, inicia uma série de reações químicas cujo produto final é a liberação de luz. A detecção da luz por um sensor permite a determinação das bases de uma sequência de DNA. Uma característica importante desta técnica é que,

Figura 2.13: Sequenciador automático Roche/454 FLX (Roche, 2009).

a cada vez que um mesmo nucleotídeo é incorporado à sequência, a intensidade da luz liberada aumenta. Se essa intensidade ultrapassar a capacidade do detector de luz, a leitura do número de bases iguais será incorreta. Este é o principal tipo de erro enfrentado por este tipo de sequenciador: a incorreta determinação do número de bases em uma cadeia com repetições seguidas do mesmo nucleotídeo, tal como CCCCCCC (Alvarez, 2009; Mardis, 2008).

O primeiro passo no processo de sequenciamento utilizando o Roche/454 FLX consiste na amplificação do DNA a ser sequenciado. Isso é feito misturando-se os fragmentos de DNA com estruturas de agarose[1] contendo sequências de DNA complementares às sequências adaptadoras do Roche/454 FLX, presentes nos fragmentos a serem sequenciados. Dessa forma, cada estrutura de agarose fica ligada a um único fragmento de DNA. A seguir, cada uma dessas estruturas contendo um fragmento de DNA é isolada em contas óleo–água contendo reagentes para a enzima DNA polimerase. Através de um ciclo térmico, são produzidas um milhão de cópias do fragmento de DNA contidos na superfície da estrutura de agarose (Alvarez, 2009; Mardis, 2008).

Após a amplificação do DNA, o sequenciamento pode ser de fato realizado. Cada estrutura de agarose é colocada em um recipiente de estrutura de sílica capilar, contendo centenas de milhares de locais para inserção de uma estrutura de agarose. O objetivo desses recipientes é fornecer uma localização fixa para monitoramento das reações de sequenciamento. Em cada recipiente, enzimas que catalizam a reação de pirosequenciamento são adicionadas a cada recipiente e a mistura é centrifu-

[1]Polímero composto de subunidades de galactose. Quando dissolvida em água quente e seguidamente arrefecida, a agarose toma uma consistência gelatinosa, este gel é muito utilizado em biologia molecular para atividades como sequenciamento.

gada com o objetivo de cobrir as agaroses com as enzimas (Alvarez, 2009; Mardis, 2008).

A incorporação de cada nucleotídeo é feita um passo por vez, e em cada passo um sensor CCD[2] registra a luz emitida em cada recipiente, assim determinando a sequência de DNA, uma base por vez. No entanto, tal sensor não consegue interpretar corretamente a incorporação de um mesmo nucleotídeo várias vezes (mais de 6), o que significa que porções de DNA nas quais uma mesma base ocorre várias vezes podem ser interpretadas, de forma equivocada, como erros no sequenciamento, tais como erros de inserção ou de remoção (Alvarez, 2009; Mardis, 2008).

O sequenciador Roche/454 FLX provê sequências de cerca de 250 bases de comprimento durante um processamento de 8 horas. Após um processamento para a remoção de sequências com baixa qualidade, são obtidas cerca de 100 milhões de bases com boa qualidade em média. Apesar do tamanho das sequências obtidas com o sequenciador Roche/454 FLX ser muito menor em comparação com os sequenciadores Sanger, o mesmo foi utilizado com sucesso no sequenciamento de genomas virais e bacteriais com alta qualidade (Alvarez, 2009; Mardis, 2008). A Figura 2.14 mostra esquematicamente o processo de sequenciamento utilizado pelo Roche/454 FLX.

Pipeline de Sequenciamento

Assim como ocorre no sequenciamento Sanger, é necessária a utilização de um sistema de Bioinformática para processar os fragmentos de DNA obtidos através dos métodos de sequenciamento de alto desempenho. Tipicamente, o *pipeline* de softwares é constituído de quatro fases: submissão, mapeamento, montagem e anotação. Particularmente, como os métodos de alto desempenho produzem sequências muito pequenas, a montagem direta de todos os fragmentos fica mais complexa. Nesse *pipeline*, os dados das sequências são obtidos do sequenciador e diretamente armazenados. Após isso, em geral, as sequências são alinhadas a um genoma de referência, formando grupos de sequências próximas. A seguir, cada um desses grupos é montado por meio de um software de montagem, obtendo assim os *singlets* e *contigs*. Finalmente, os *singlets* e *contigs* obtidos anteriormente são anotados (Alvarez, 2009; Mardis, 2008).

Submissão

Ao contrário dos dados de sequenciadores Sanger, o sequenciador Roche/454 FLX não provê dados que permitam que bases individuais possam ser determinadas. Ao invés disso, estima-se o comprimento de cada homopolímero na sequência. Por exemplo, a sequência AAATGGC seria armazenada como constituída de uma sequência de 3 A's, seguida de uma sequência de 1 T, uma sequência de 2 G's e, por fim, uma sequência de um único C. A determinação da sequência consiste em simples-

[2]Sigla para *Charge-Coupled Device* — Dispositivo de Carga Acoplada. Trata-se de um sensor para captação de imagens formado por um circuito integrado contendo uma matriz de capacitores ligados (acoplados). É comumente utilizado em câmeras digitais.

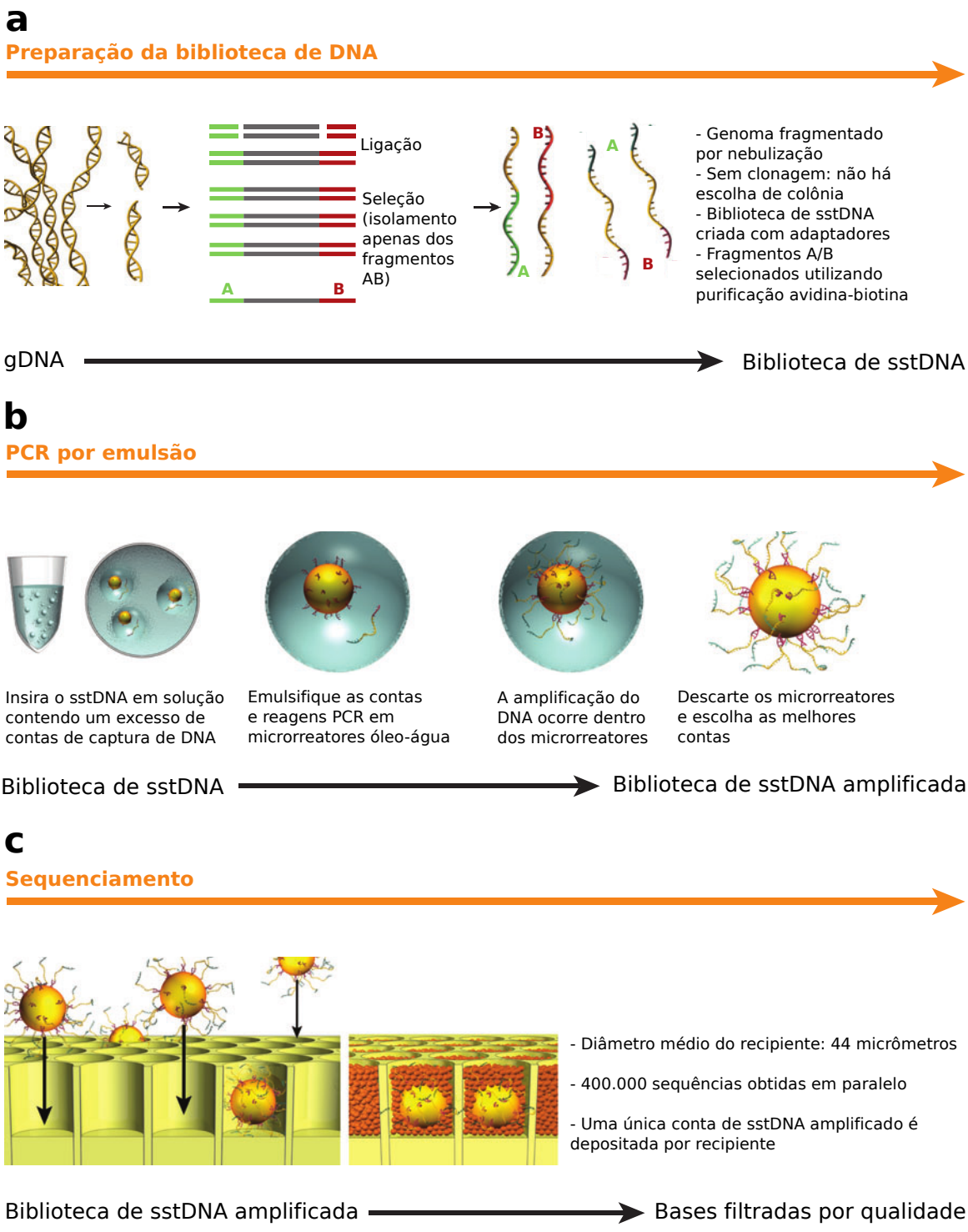


Figura 2.14: Método utilizado pelo sequenciador Roche/454 FLX. Adaptado de Mardis (2008).

mente analisar as estimativas do sequenciador e concatenar os diversos homopolímeros determinados (Alvarez, 2009).

Os dados do sequenciador Roche/454 FLX são disponibilizados em arquivos binários no formato *Standard Flowgram File* (SFF), os quais podem ser processados pelo programa Flower (BIOHASKELL, 2010), capaz de produzir arquivos de sequência e de qualidade em formato *FASTA*. A partir daí, as sequências podem ser armazenadas num banco de dados, tal como é feito em projetos de sequenciamento Sanger.

Mapeamento

Uma vez que as sequências obtidas pelos novos sequenciadores são relativamente curtas em relação ao sequenciamento Sanger, isso torna complexo o uso das técnicas tradicionais para reagrupar os fragmentos sequenciados no DNA original. De qualquer forma, é desejável aplicar as técnicas antigas aos novos dados, mesmo sendo necessário efetuar adaptações (Alvarez, 2009).

Uma possível abordagem seria usar um genoma de referência, normalmente um organismo semelhante àquele que está sendo sequenciado, cujo genoma já fosse bem conhecido. Assim, é possível mapear as pequenas sequências obtidas pelos novos sequenciadores sobre o genoma bem conhecido, agrupando-as conforme suas posições no mapeamento. Uma vez que as sequências agrupadas constituem um número muito menor a ser analisado e possuem poucas diferenças entre si, uma vez que estão mapeadas aproximadamente na mesma região do genoma, seria possível aplicar técnicas de montagem tradicional a esses grupos de sequências (Alvarez, 2009).

O programa Maq (Li et al., 2008) é capaz de realizar essa tarefa de mapeamento. A técnica do software consiste em fragmentar cada sequência a ser mapeada em quatro pedaços menores, de tamanho aproximadamente igual, chamados de sementes. Dado que uma sequência só se alinhará perfeitamente ao genoma de referência se todas as sementes alinharem-se perfeitamente — se houver alguma diferença, esta deverá estar contida em uma semente —, pode-se procurar todos os possíveis locais nos quais a sequência pode se alinhar ao genoma, permitindo no máximo duas diferenças. Com base nesse espaço reduzido de locais, pode ser feita a busca da sequência nesses locais e reportar o mapeamento ao usuário (Alvarez, 2009).

Montagem

Uma vez obtidas as sequências mapeadas, é possível aplicar as técnicas de montagem tradicionais do sequenciamento Sanger, com pequenas modificações. Um dos softwares para montagem de sequências que pode ser utilizado é o CABOG (Miller et al., 2008), que consiste numa adaptação de um montador para sequenciamento Sanger desenvolvido pela Celera.

Anotação

A anotação de sequências é realizada para sequenciamentos de alto desempenho de forma análoga ao aplicado ao método Sanger. Também são utilizados softwares de alinhamento de sequências, tais como o BLAST, a fim de buscar similaridades entre as sequências do organismo em estudo e as sequências de organismos bem estudados.

Outros Métodos de Sequenciamento de Alto Desempenho

Além do Roche/454 FLX, outros sequenciadores de alto desempenho foram desenvolvidos. É o caso do Illumina Genome Analyzer e do Applied Biosystems SOLiD^{TM}(Mardis, 2008). Embora cada um desses sequenciadores utilize métodos diferentes para a obtenção das cadeias genômicas, ambos apresentam desafios semelhantes para os sistemas de Bioinformática, tal como apresentado para o Roche/454 FLX, uma vez que produzem sequências de tamanho extremamente reduzido.

2.7 Genômica Comparativa

A genômica comparativa é uma técnica de estudo de como dados genômicos, tais como localização de cadeias e genes, funções e categorias, de diferentes espécies estão relacionados (Bachhawat, 2006). Através da genômica comparativa tem sido possível aplicar descobertas já feitas a espécies que estão sendo estudadas, baseado nas pesquisas que já foram realizadas em organismos que foram previamente sequenciados e analisados. As aplicações mais comuns da genômica comparativa são a descoberta de genes e de RNAs não codificadores.

Essa técnica está fortemente relacionada com a evolução das espécies, sendo largamente utilizada para estabelecer relações evolutivas entre diferentes organismos. Geralmente, o estudo das relações evolutivas entre as espécies é feito utilizando-se árvores filogenéticas. Esse estudo é feito com o auxílio de softwares de visualização que mostram as relações de parentesco entre os organismos analisados. Visualizadores de árvores filogenéticas e outros softwares utilizados em tarefas de genômica comparativa são ilustrados na Seção 3.1.

Com o crescimento dos bancos de dados públicos de anotações de genomas sequenciados, a genômica comparativa tem sido cada vez mais utilizada nos projetos de sequenciamento para a determinação de funções e categorias das sequências obtidas, tal como descrito no tópico **Anotação** da Subseção 2.6.1. A abordagem computacional mais comum é o alinhamento textual de sequências, com a utilização de programas como o BLAST. Porém, o volume crescente de dados faz surgir a necessidade de ferramentas de visualização que facilitem as análises feitas pelos pesquisadores, uma vez que programas como o BLAST geram resultados em forma de texto, com muitas informações detalhadas (ver Seção 3.2).

Outra tarefa de genômica comparativa é a identificação de sintenias entre diferentes espécies.

2.7.1 Sintenias

A palavra *sintenia* deriva do termo *synteny* da língua inglesa, que trata-se de um neologismo com o significado de "on the same ribbon" ("na mesma fita"). Esse novo termo foi proposto por John H. Renwick em 1971. Em sua concepção original, *sintenia* refere-se à localização de genes no mesmo cromossomo (Passarge et al., 1999).

Neste trabalho, um par de genes é dito sintênico quando estes se conservam dentro de um mesmo cromossomo, *supercontig* ou *scaffold* entre espécies diferentes. Um *supercontig* é um conjunto ordenado e orientado de *contigs* que ainda contém alguns *gaps*. Um *scaffold* é um conjunto de *contigs* que já se aproxima da estrutura de um cromossomo.

A identificação de sintenias entre diferentes espécies necessariamente decorre de comparações entre múltiplos genomas, estando fortemente relacionada às técnicas clássicas de genômica comparativa. Em geral, a identificação de sintenias parte da comparação do genoma em estudo com outros genomas bem conhecidos, nos quais se buscam a ocorrência de genes sintênicos ao do organismo pesquisado. Tais comparações comumente são feitas utilizando softwares de alinhamento de sequências, tais como o BLAST. O passo seguinte frequentemente consiste em utilizar alguma ferramenta de visualização gráfica dos resultados dos alinhamentos obtidos, de modo a facilitar a identificação de sintenias.

Na Seção 3.3 são apresentados e comparados os softwares de visualização mais utilizados para a identificação de sintenias. Em seguida, na Seção 4.2 argumenta-se que ainda há espaço para o desenvolvimento de uma nova ferramenta de visualização, com uma abordagem diferente das demais.

Capítulo 3

Visualização de Dados Biológicos

Com a expansão dos projetos de sequenciamento de genomas pelo mundo, o volume de dados produzido tem crescido enormemente. Somente o banco de dados do GenBank tem dobrado a quantidade de dados a cada 18 meses, desde 1982, ultrapassando 95 bilhões de pares de bases (Lathe et al., 2008). Um volume tão grande de dados demanda ferramentas de visualização e análise intuitivas, eficazes e eficientes. Um projeto de sequenciamento de genoma produz os mais variados tipos de dados: alinhamentos e montagens de sequências, referências textuais e muitos *links* com outros bancos de dados.

A Seção 3.1 apresenta o problema da visualização de dados biológicos e os modos de visualização mais comuns. Na Seção 3.2 são apresentados os requisitos mais comuns das ferramentas utilizadas para a visualização de comparações de sequências, com vistas à identificação de sintenias. Em seguida, a Seção 3.3 relaciona as ferramentas mais comuns para a visualização de sintenias e um quadro comparativo.

3.1 Dados Biológicos e sua Visualização: Gráficos Mais Comuns

O maior volume de dados biológicos certamente diz respeito ao sequenciamento de genomas. Nesse contexto, são armazenados dados sobre cada par de base de um organismo, sequências de interesse que foram identificadas, sobretudo genes, com suas respectivas anotações e *links* para outras bases de informação. O principal exemplo de visualizador capaz de integrar a maior parte dos dados gerados por projetos de sequenciamento é o NCBI Map Viewer, um visualizador de dados do GenBank (Sayers et al., 2010). A Figura 3.1 ilustra como os dados são apresentados pelo Map Viewer.

Outra informação muito importante para os biólogos, e que muitas vezes deriva dos dados gerados pelos projetos de sequenciamento, é a reconstrução das relações evolucionárias entre as espécies, o que é conhecido como filogenética. Nesse sentido, os gráficos produzidos para melhor compreensão dessas relações evolucionárias é uma "árvore da vida", um gráfico conhecido como árvore filogenética, que busca evidenciar o grau de parentesco entre diferentes espécies. As primei-

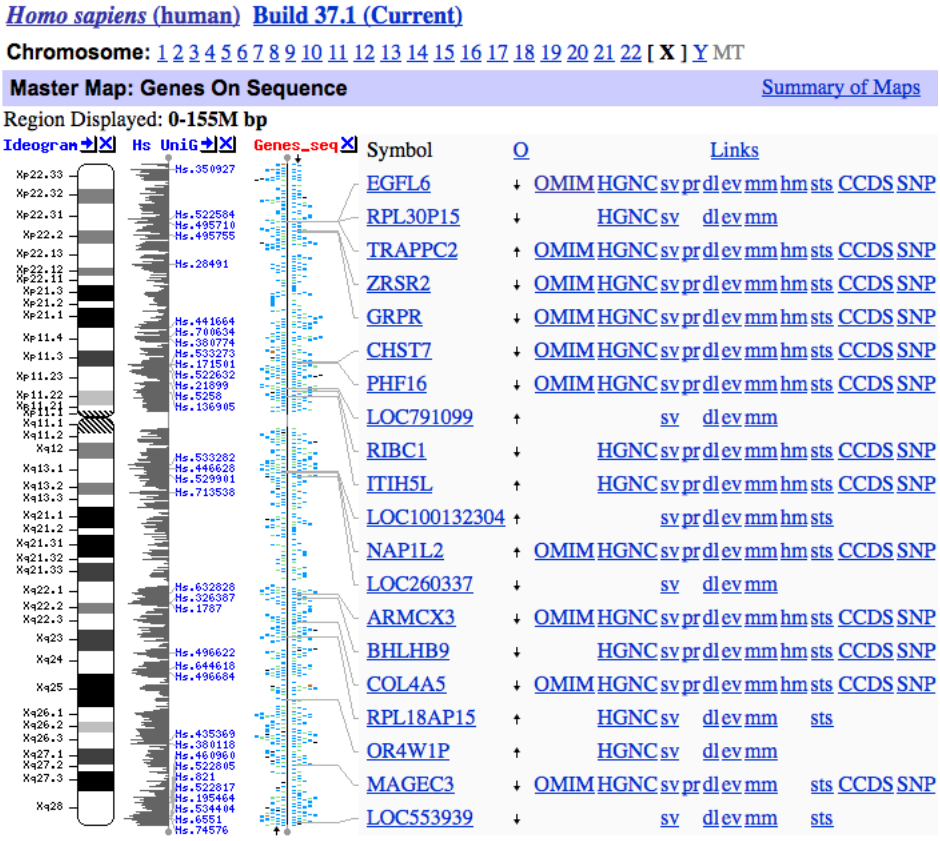


Figura 3.1: Visualização do cromossomo X humano pelo NCBI Map Viewer (NCBI, 2010). Uma característica importante é a possibilidade de acessar a anotação de um gene mapeado através de *links*.

ras ferramentas de visualização de árvores filogenéticas datam de 1996, quando apresentavam simples arestas que ligavam as folhas das árvores a uma raiz (Page, 1996). Os aplicativos mais recentes, porém, fornecem gráficos mais complexos, fornecendo interatividade e *links* para dados das anotações (Letunic and Bork, 2007), conforme ilustrado pela Figura 3.2.

Outro tipo de informação muito relevante para os estudos sobre um determinado organismo é aquela relacionada à dinâmica das reações químicas que ocorrem no interior das células: o metabolismo. Nesse sentido, é comum a elaboração de gráficos denominados vias metabólicas, que apresentam toda a cadeia de reações químicas de determinados processos celulares. Uma das fontes de gráficos de vias metabólicas é o banco de dados do KEGG, constituído de ilustrações feitas à mão, no qual é possível efetuar consultas sobre os produtos de genes e outras moléculas envolvidas (Kanehisa et al., 2006). A Figura 3.3 apresenta uma via metabólica disponibilizada pelo KEGG.

Finalmente, a comparação entre sequências demanda também a elaboração de ferramentas de visualização avançadas, o que será abordado na próxima seção.

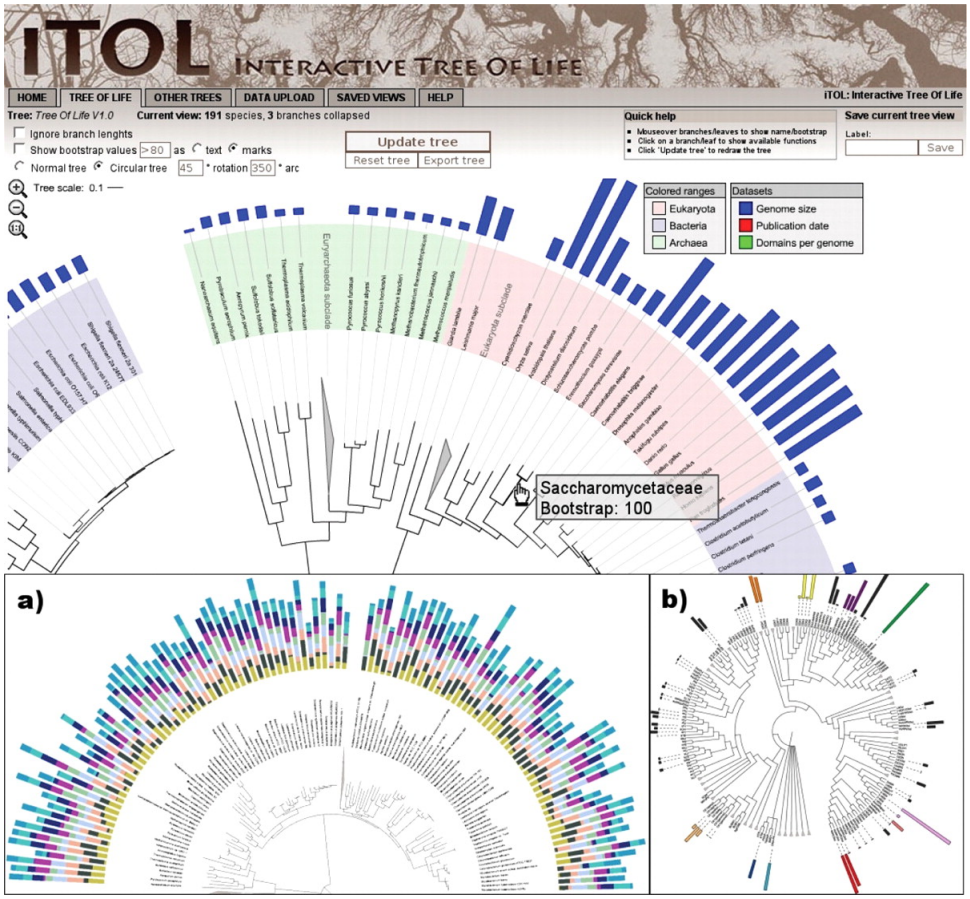


Figura 3.2: Árvores filogenéticas apresentadas pelo software ITOL — *Interactive Tree Of Life* (Letunic and Bork, 2007).

3.2 Comparação de Sequências e Identificação de Sintenias

A comparação de sequências é uma das atividades mais comuns na pesquisa relacionada ao sequenciamento de genomas. As ferramentas de comparação de sequências mais populares datam de 1984, com o FASTP, que compara cadeias de aminoácidos (Lipman and Pearson, 1985), seguido por pacotes de programas de comparação também de cadeias de nucleotídeos, como o FASTA (Pearson and Lipman, 1988) e o BLAST (Altschul et al., 1990), até softwares mais recentes, como o BLAT (Kent, 2002). Todos esses programas tornaram-se bastante populares entre os pesquisadores e são fundamentais para as pesquisas genômicas, sobretudo durante a fase de anotação de um genoma.

O volume de dados produzidos por esses programas de comparação de sequência é enorme e tem crescido exponencialmente (Lathe et al., 2008). Porém, a análise dos dados gerados por esses programas não é uma tarefa trivial. Em geral, a quantidade de sequências comparadas de uma só vez é muito grande, o que resulta em arquivos de resultado bastante extensos. Como os programas FASTA, BLAST e BLAT produzem como resultado arquivos textuais, tal como ilustra a Figura 3.4,

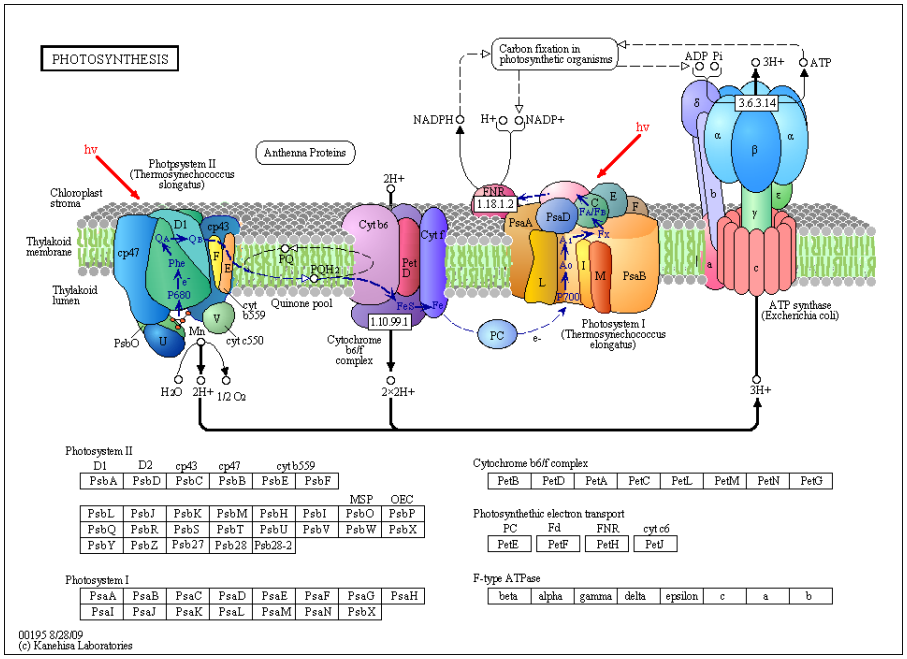


Figura 3.3: Exemplo de via metabólica disponibilizada pelo KEGG, ilustrando o processo de fotossíntese (KEGG, 2010).

identificar características globais do genoma, como localização relativa de genes em cromossomos, não é uma tarefa das mais fáceis.

Em especial, a tarefa de identificar sintenias tem sido feita pelos biólogos com o auxílio de programas de comparação de sequências, na maioria dos casos, tem-se utilizado o BLAST. Contudo, identificar sintenias requer a caracterização e localização de grupos de genes nos cromossomos de um organismo e como esses grupos conservam o posicionamento relativo de seus genes no genoma de um outro organismo. Os programas FASTA, BLAST e BLAT fornecem as coordenadas precisas de onde uma determinada cadeia (possivelmente um gene) começa e termina num genoma. Porém, como ilustrado pela Figura 3.4, esses programas produzem um resultado que é de difícil análise pelos pesquisadores. Então, faz-se necessária o desenvolvimento de ferramentas de visualização dos resultados gerados por esses programas, de modo a prover uma visão mais clara das comparações efetuadas.

A seguir são apresentadas as características que os softwares de visualização devem ter para auxiliar os biólogos na tarefa de identificar sintenias.

3.2.1 Requisitos de um Software de Visualização para a Identificação de Sintenias

Diante da dificuldade de se analisar dados apresentados de forma textual, as ferramentas de visualização precisam ser desenvolvidas tendo em vista as necessidades dos biólogos nas pesquisas sobre sintenia. Além disso, esses softwares precisam ser intuitivos e fornecer um certo grau de interatividade, de modo que o pesquisador consiga navegar facilmente pelos resultados e assim encontrar a informação que precisa. Nesse sentido, alguns trabalhos se dedicaram a elucidar quais caracterís-

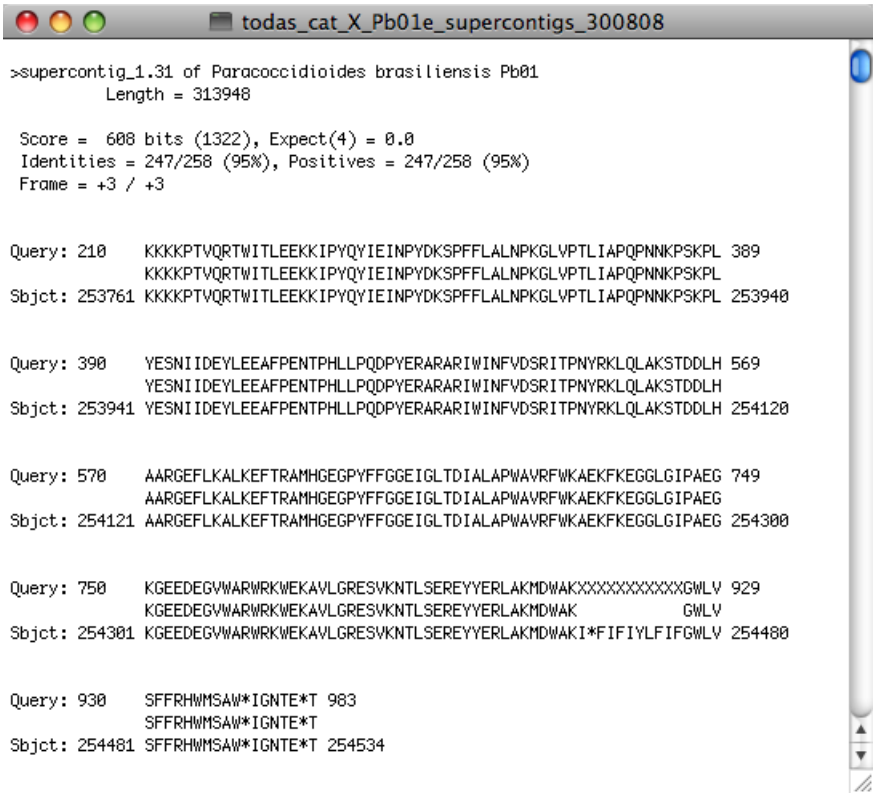


Figura 3.4: Saída gerada pelo BLAST. Como o resultado gerado é um arquivo texto, não é uma tarefa simples extrair informação sobre cromossomos ou genes.

ticas uma ferramenta de visualização de comparações de genomas deve ter, a fim de permitir a identificação de sintenias.

O trabalho de Hunt et al. (2004) aborda quais características visuais uma aplicação precisa ter para facilitar a identificação de sintenias. Esse trabalho culminou na implementação do software SyntenyVista, que será apresentada mais adiante. O artigo inicialmente enumera quais são os desafios que toda ferramenta de visualização de sintenias deveria solucionar, a saber:

- **Visão de todo o cromossomo**: é útil para o usuário ter uma visão global de todas as relações entre um cromossomo e outros possíveis. O problema é que o volume de informação pode confundir facilmente o usuário.
- **Todo o cromossomo, com detalhe**: o usuário deve ser capaz de conseguir focar uma determinada região de um cromossomo, enquanto ainda tem à disposição uma visão do todo. Mais uma vez o problema é o grande volume de informação.
- **Escala**: devido à grande extensão de um cromossomo inteiro, em contraste com o tamanho limitado de um único gene, é preciso estabelecer algum critério de escala, de modo que seja possível apresentar essas duas informações lado a lado.

- **Orientação do mapa**: gráficos horizontais são mais fáceis de navegar, contudo apresentam dificuldades para a colocação dos rótulos dos objetos apresentados.
- **Problema de rotulação**: a área de exibição em geral é muito pequena e o volume de dados a serem apresentados é muito grande.
- **Representação das relações entre os objetos**: este é o foco da visualização de sintenias. As relações podem ser evidenciadas pelo traçado de linhas ou trapézios ou alinhando os objetos sobre uma grade. As linhas deixam as relações mais legíveis, mas são desnecessárias quando não há inversão de ordem de genes, situação em que é mais desejável o traçado de trapézios.
- **Cruzamento de linhas**: a inversão na ordem de genes resultam em cruzamentos de linhas, o que dificulta a legibilidade. É desejável minimizar o cruzamento de linhas a fim de facilitar a legibilidade do gráfico.
- **Plano de fundo**: a cor e textura devem facilitar a legibilidade, sobretudo em contraste às linhas que representam as sintenias.

Diante desses desafios, Hunt et al. (2004) elaboraram uma lista das funcionalidades que visualizadores de sintenias devem apresentar:

A) **Detalhes sob demanda**: o usuário deve ser capaz de selecionar as espécies, cromossomos e áreas do cromossomo que deseja visualizar.

B) ***Zoom***: é necessário para a visualização de grande objetos em áreas restritas.

C) **Rotulação eficiente**: os rótulos dos objetos devem ser nítidos e não devem se sobrepor.

D) **Movimentar um cromossomo ao longo do seu eixo**: deve ser possível deslizar um cromossomo ao longo de seu eixo, de modo a permitir o alinhamento de blocos sintênicos e, assim, facilitar a visualização.

E) **Aplicar escala ao cromossomo**: deve ser implementado algum mecanismo que compacte a área de exibição dos objetos, de modo a tornar a visualização mais compacta e legível.

F) **Inversão do cromossomo**: isto permite que o usuário veja mais claramente as relações de sintenia em situações de inversão de ordem dos genes.

G) **Filtragem**: onde há muitos dados para serem exibidos, deve haver a possibilidade de filtrar os dados de acordo com algum tipo ou outra característica.

H) **Coloração**: o software deve usar um esquema de coloração por padrão para diferenciar cromossomos e genes, mas deve permitir que o usuário modifique as cores.

Por outro lado, existe uma série de características que um pesquisador espera descobrir ao analisar uma visualização de sintenias entre dois ou mais genomas comparados. Meyer et al. (2009) elaboraram uma lista das questões que um pesquisador espera esclarecer ao estudar os gráficos de comparações de genomas, no

contexto da visualização de sintenias. Segundo os autores, as características que um pesquisador tenta elucidar ao utilizar um visualizador de sintenias são:

1. Quais cromossomos compartilham blocos de genes conservados.
2. Para um cromossomo, com quantos outros cromossomos ele compartilha blocos de genes conservados.
3. Qual a densidade de cobertura do genoma e onde estão os *gaps*.
4. Onde estão os blocos de genes, se estariam em torno de uma posição específica no cromossomo.
5. Quais são os tamanhos e localizações de outras características genômicas próximas a um bloco de genes.
6. Quão grandes são os blocos de genes.
7. Se blocos de genes vizinhos se conservam no mesmo cromossomo e/ou preservam seu posicionamento relativo.
8. Se a orientação dos pares de blocos de genes é preservada ou invertida.
9. Se a orientação é preservada para blocos de genes vizinhos.
10. Se as pontuações de similaridades são iguais com respeito a blocos de genes vizinhos.
11. Se os genes pareados dentro de um bloco são contíguos.
12. Quão grande é um gene em relação a outros genes dentro de um bloco.
13. Quais são os tamanhos, localizações e nomes dos genes dentro de um bloco.
14. Quais são as diferenças entre nucleotídeos individuais e pares de genes.

Diante dessas questões, Meyer et al. (2009) desenvolveram um visualizador denominado MizBee, que será tratado em detalhes mais adiante.

3.3 Visualizadores de Comparações de Sequências para a Identificação de Sintenias

Nesta Seção são apresentadas algumas das ferramentas mais comuns para visualização de genomas baseadas em comparações de sequências de nucleotídeos. Algumas delas também apresentam uma visualização baseada na exibição da estrutura física dos cromossomos, ou seja, mostram onde os genes estão localizados nos cromossomos.

Em geral, essas ferramentas recebem como entrada sequências montadas de nucleotídeos e utilizam algum software de comparação, como o BLAST. No entanto, alguns visualizadores recebem como entrada dados pré-processados sobre a localização dos genes em cada cromossomo. A tarefa de determinar se um par

de sequências é sintênico em geral é de responsabilidade do usuário, embora existam ferramentas capazes de pré-determinar genes sintênicos entre os organismos comparados.

O modo de visualização das ferramentas varia bastante. Meyer et al. (2009) compilaram um quadro comparativo dos modos de visualização mais comuns, conforme ilustra a Figura 3.5. Certamente, o modo de visualização mais comum é o *linear separado*, mas há softwares que combinam diversos modos de visualização, a depender do nível de detalhes escolhido pelo usuário.

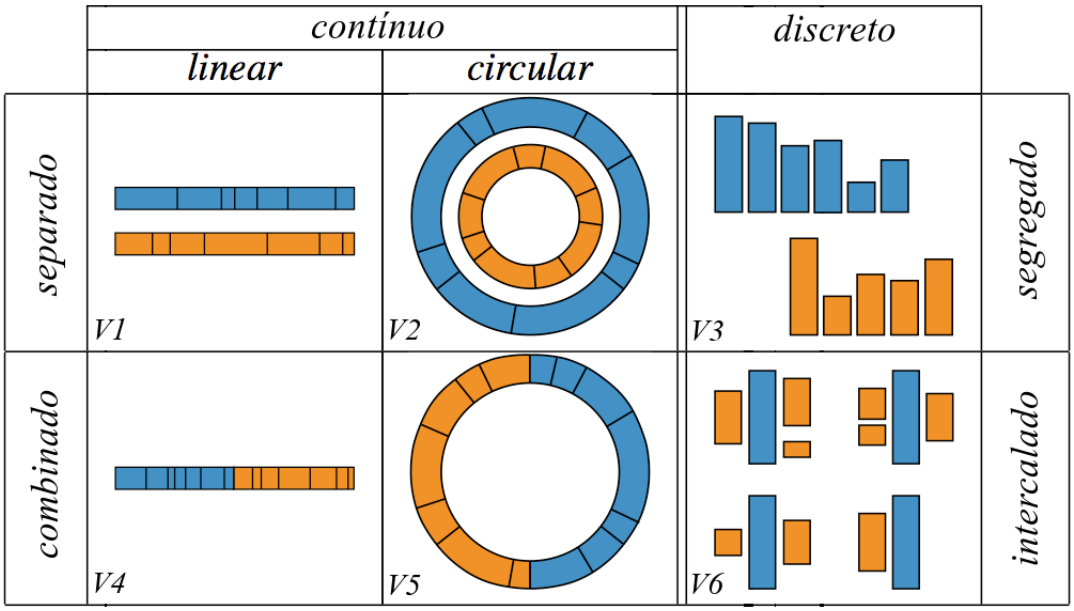


Figura 3.5: Tipos mais comuns de gráficos usados para visualização de sintenias. Em azul o genoma de origem e em laranja o genoma de destino. Adaptado de Meyer et al. (2009).

3.3.1 Softwares para Visualização de Sintenias

A seguir são apresentadas as ferramentas mais comuns para a visualização de sintenias.

Apollo

Apollo é uma ferramenta de apoio à anotação em geral, que permite a identificação de sintenias entre dois genomas (Lewis et al., 2002). Trata-se de um aplicativo *desktop*, escrito em Java, que utiliza o BLAST para processar as comparações. Os genes ortólogos entre os dois genomas são ligados por trapézios, conforme ilustrado pela Figura 3.6.

SyntenyView

SyntenyView é um módulo do Ensembl para a visualização de sintenias entre genomas de dois organismos (Clamp et al., 2003). O projeto Ensembl consiste de um

banco de dados que provê um *framework* de Bioinformática para organizar grandes volumes de dados biológicos em torno de sequências de extensos genomas. Ele é uma fonte estável de anotações sobre sequências dos genomas humano, do rato e de outras espécies, disponível por meio de uma página web interativa ou por arquivos texto que podem ser baixados pelo usuário. A ferramenta SyntenyView utiliza BLAST para comparações entre nucleotídeos ou entre aminoácidos, apresentando uma visualização da estrutura do cromossomo (Figura 3.7).

SyntenyVista

SyntenyVista é um aplicativo *desktop*, escrito em Java, para a visualização de sintenias entre dois genomas (Hunt et al., 2004). A ferramenta apresenta a comparação de um par de cromossomos por vez, com a possibilidade de exibir os genes tal como estão dispostos fisicamente, assim como apresentando pelo SyntenyView, ou então pode apresentá-los de forma compacta, preservando somente sua ordem, o que os autores denominam como *cartoon scaling*. No modo de *cartoon scaling*, as distâncias entre os genes seriam desprezadas, assim como seu tamanho, de modo que todos sejam representados com o mesmo tamanho, o menor possível para a exibição de um rótulo. SyntenyVista permite que o usuário inverta um cromossomo, de modo a reduzir o cruzamento das linhas que ligam os genes dos dois organismos (Figura 3.8), além de oferecer diversos níveis de *zoom*. O software ainda é capaz de se conectar ao banco de dados do Ensembl para recuperar informações detalhadas sobre as anotações dos genes.

Mauve

Mauve é um software de alinhamento múltiplos de genomas, com algoritmos otimizados para o caso em que há conservação de genes entre as espécies estudadas, e que fornece uma interface gráfica para visualização dos genomas comparados (Darling et al., 2004). A visualização disponibilizada pela ferramenta consiste em apresentar os genes conforme estão dispostos no genoma, ligando-os aos respectivos alinhamentos no genoma seguinte (Figura 3.9). A ferramenta também utiliza uma técnica de gradação de cores para representar o grau de similaridade entre os blocos.

ACT

ACT é um aplicativo *desktop*, escrito em Java, com a finalidade de exibir graficamente comparações de múltiplas sequências (Carver et al., 2005). A comparação é feita par a par, utilizando o programa BLAST, sendo capaz de exibir os dados por este gerados. A exibição das comparações é feita com o traçado de trapézios delimitados pelas cadeias que são semelhantes entre um par de genomas (Figura 3.10). A coloração dos trapézios é feita de acordo com o grau de similaridade dos cadeias de nucleotídeos.

SynBrowse

SynBrowse é um módulo que funciona integrado ao GBrowse (Pan et al., 2005). O Generic Genome Browser (GBrowse) é uma combinação de banco de dados e interface web interativa para a manipulação e exibição de anotações em genomas (Stein et al., 2002). A ferramenta disponibiliza a visualização de comparações de sequências de dois organismos, provendo modos de exibição que facilitam a identificação de macrosintenias, microsintenias e genes homólogos (Figura 3.11).

A ferramenta utiliza também o BLAST para efetuar as comparações. SynBrowse é capaz de determinar sequências sintênicas, procurando por pares de genes (ou alinhamentos) que ocorram na mesma ordem em ambas sequências, numa distância menor do que um limite definido pelo usuário. Assim, um conjunto igual ou maior que um número mínimo (especificado pelo usuário) de tais pares de genes é considerado um bloco de sintenia, considerando parâmetros como a qualidade do alinhamento e colinearidade das sequências. A ferramenta ainda é capaz de exibir o alinhamento textual original (gerado pelo BLAST), além de os gráficos gerados possuírem elementos clicáveis, capazes de direcionar o usuário a informações mais detalhadas sobre a anotação. O software é escrito em Perl.

SynView

SynView é uma ferramenta baseada no GBrowse e distribuída como parte deste (Wang et al., 2006). SynView permite a visualização da comparação entre múltiplos genomas, baseada na escolha de um genoma de referência pelo usuário. É capaz de exibir tanto a comparação quanto as informações de anotação. Assim como no ACT, a comparação é feita par a par, utilizando o programa BLAST. A exibição das comparações também é feita com o traçado de trapézios e a coloração feita de acordo com o grau de similaridade dos cadeias de nucleotídeos. Além disso, os trapézios podem conter links para páginas de descrição. A ferramenta é escrita em Perl.

GBrowse_syn

GBrowse_syn é mais um módulo para visualização de sintenias (Figura 3.13) integrado ao GBrowse (McKay, 2007). Permite a visualização de comparações entre múltiplas espécies.

Cinteny

Cinteny é um aplicativo web, escrito em C++ e PHP para a visualização de sintenias entre múltiplos genomas (Sinha and Meller, 2007). O software utiliza uma árvore de busca ternária para representar os genomas, com os genes representados pelas folhas. Cinteny é capaz de identificar sintenias calculando a distância reversa entre grupos de genes. A ferramenta possui três níveis de visualização: todo o genoma, cromossomo ou por genes individuais (Figura 3.14). Cinteny ainda é capaz de fazer de se conectar com o NCBI para fornecer informações mais detalhadas sobre os genes exibidos.

MEDEA

MEDEA é uma ferramenta que exibe sequências de múltiplas espécies para visualização de sintenias (Broad Institute, 2009). O software foi desenvolvido em ActionScript 3, provendo diversos níveis de visualização (Figura 3.15). Os dados precisam ser previamente formatados para a visualização dos genomas, ou seja, a ferramenta funciona somente como um visualizador.

Sybil

Sybil é um aplicativo web (Figura 3.16), escrito em Perl, capaz de efetuar genômica comparativa entre múltiplas espécies (TIGR, 2009). O software utiliza bancos de dados Chado, que é o modelo de banco de dados relacional utilizado pelo GBrowse, e usa o *TIGR workflow engine* para executar os programas de análise dos dados, tais como o BLAST. Sybil utiliza BLASTP (bidirecional) para comparações entre sequências de aminoácidos, sendo capaz de identificar blocos de genes sintênicos. A ferramenta ainda pode gerar gráficos nos formatos PNG, JPEG, SVG e PDF. Particularmente, Sybil apresenta a comparação entre múltiplos genomas com o objetivo de identificar sintenias num gráfico denominado *gradiente de sintenia*. Nesse gráfico, o genoma de referência é representado por uma barra colorida como um gradiente entre duas cores. Em seguida, os demais genomas são apresentados como barras horizontais do mesmo tamanho, coloridas de acordo com a cor correspondente ao apresentado no genoma de referência. Dessa forma, a conservação das cores como gradiente evidencia regiões de conservação genômica, com a provável ocorrência de sintenias.

MizBee

MizBee é uma ferramenta de visualização de sintenias entre dois genomas (Meyer et al., 2009). Trata-se de um aplicativo *desktop* escrito na linguagem Processing — um ambiente de desenvolvimento e execução de aplicações que necessitam de gráficos avançados e interativos, porém, de simples implementação, baseado em Java (Reas and Fry, 2006). A ferramenta também possui três níveis de visualização, combinando diversos tipos de gráficos (Figura 3.5). Os três níveis são exibidos simultaneamente na janela principal do aplicativo: todo o genoma, cromossomo ou por blocos de genes (Figura 3.17). Dessa forma, a visualização por cromossomo e por bloco de genes corresponde à ampliação da seleção efetuada na visualização de todo o genoma. MizBee não é capaz de identificar sintenias, funcionando apenas como um visualizador de dados previamente formatados.

3.3.2 Comparação entre as Ferramentas

A Tabela 3.1 apresenta um comparativo entre as ferramentas apresentadas previamente, com relação aos requisitos levantados na Seção 3.2.

A ferramenta que implementa o maior número de requisitos, tanto aqueles definidos por Hunt et al. (2004), quanto os definidos por Meyer et al. (2009) é o SyntenyVista. Contudo, esse software é capaz de comparar somente dois genomas.

Dentre aquelas ferramentas capazes de comparar múltiplos genomas, Mauve é a que possui o conjunto mais equilibrado de funcionalidades. Assim, caso o pesquisador precise comparar somente dois genomas, SyntenyVista é o software mais adequado. Mas se é necessário comparar múltiplos genomas, a ferramenta a ser utilizada é Mauve.

Portanto, apesar de já terem sido desenvolvidas diversas ferramentas que auxiliem na identificação de sintenias, ainda há espaço para o projeto e implementação de um novo software, que combine recursos visuais que simplifiquem a análise feita pelo pesquisador, associados à capacidade de comparar múltiplos genomas. Nesse sentido, este trabalho apresenta uma nova ferramenta, apresentada no Capítulo 5.

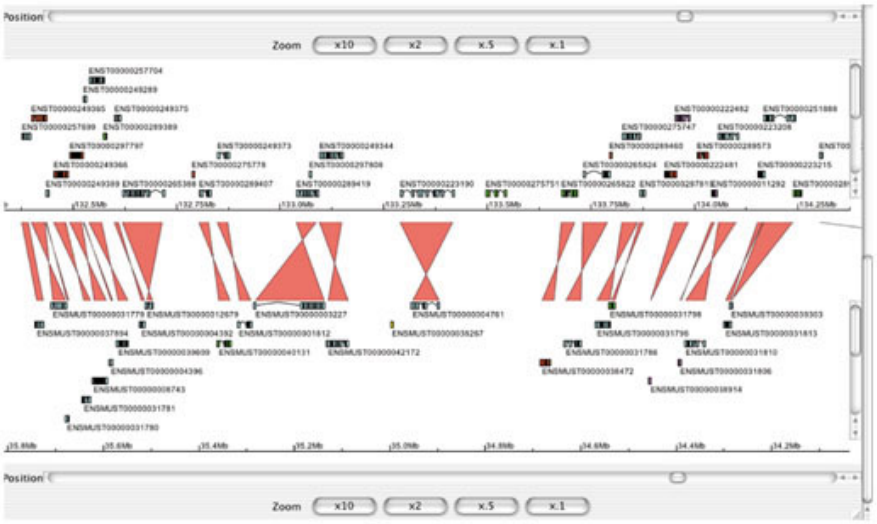


Figura 3.6: Uma visão detalhada de sintenias no Apollo.

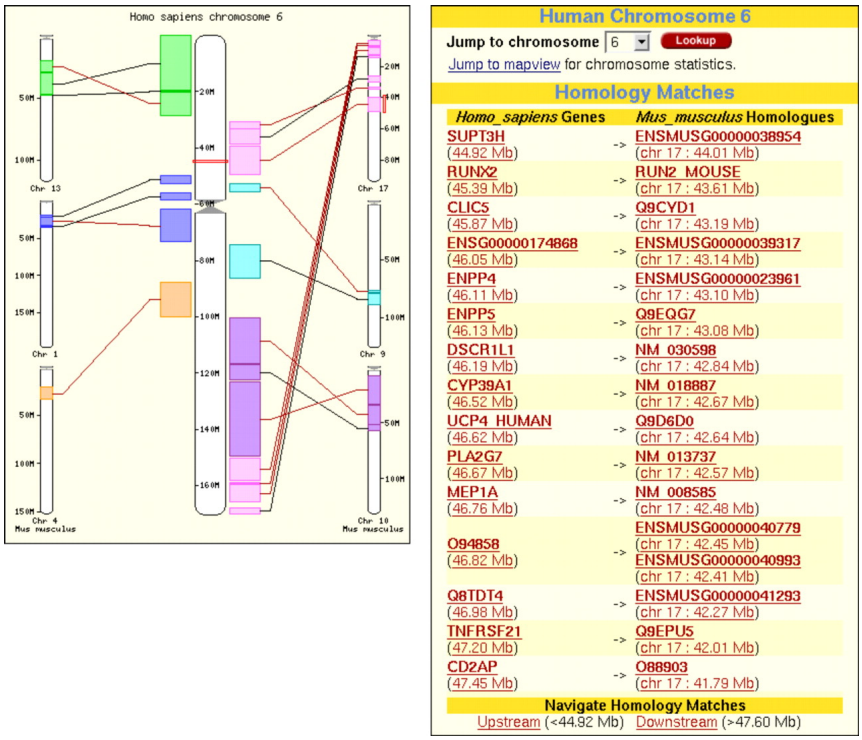


Figura 3.7: Exemplo da visualização disponibilizada pelo SyntenyView.

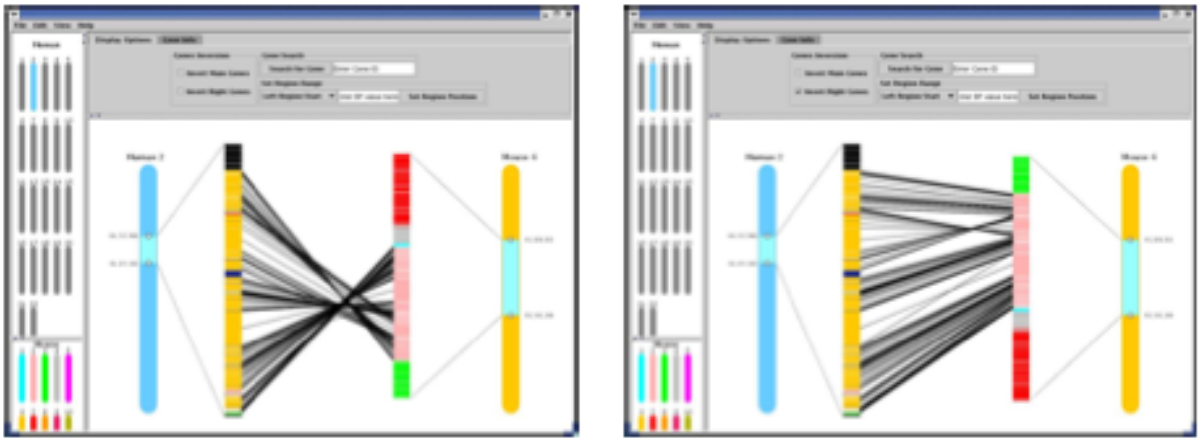


Figura 3.8: Um dos modos de visualização do SyntenyVista, destacando a inversão de um cromossomo (na janela à esquerda o cromossomo esquerdo está em posição normal, enquanto na janela à direita o mesmo cromossomo foi invertido).

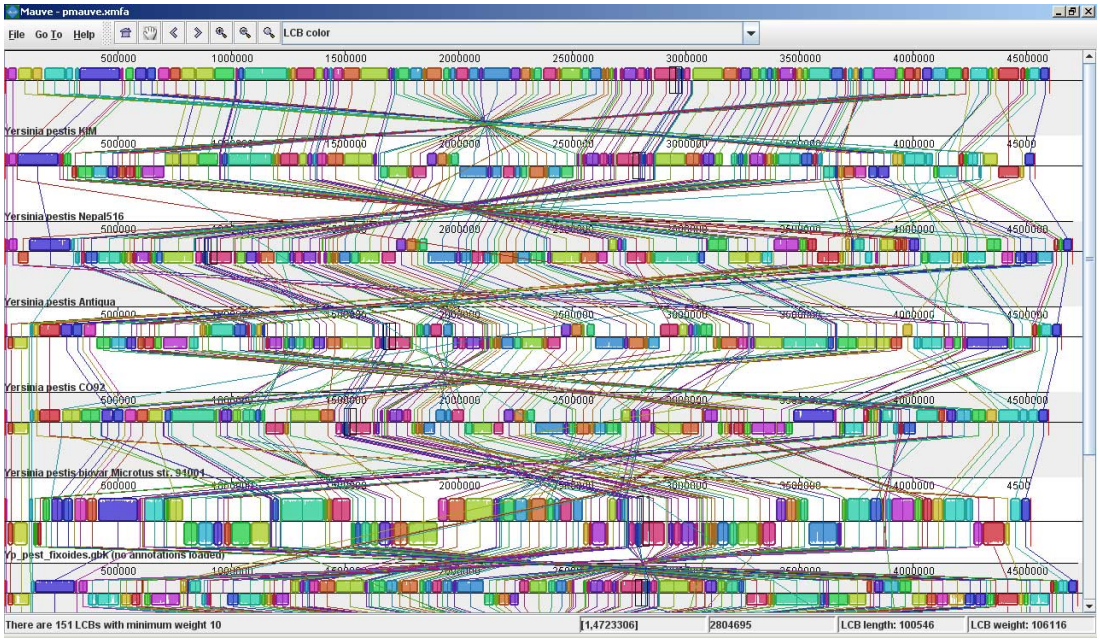


Figura 3.9: Visualização de múltiplos genomas pelo Mauve.

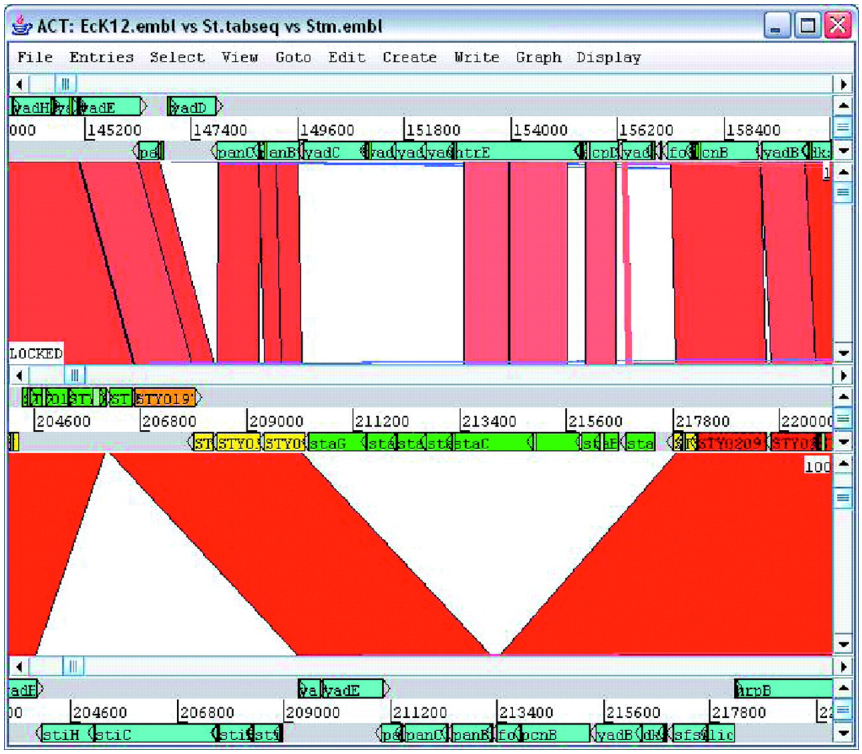


Figura 3.10: Janela principal do ACT, exibindo a visualização da comparação de três genomas.

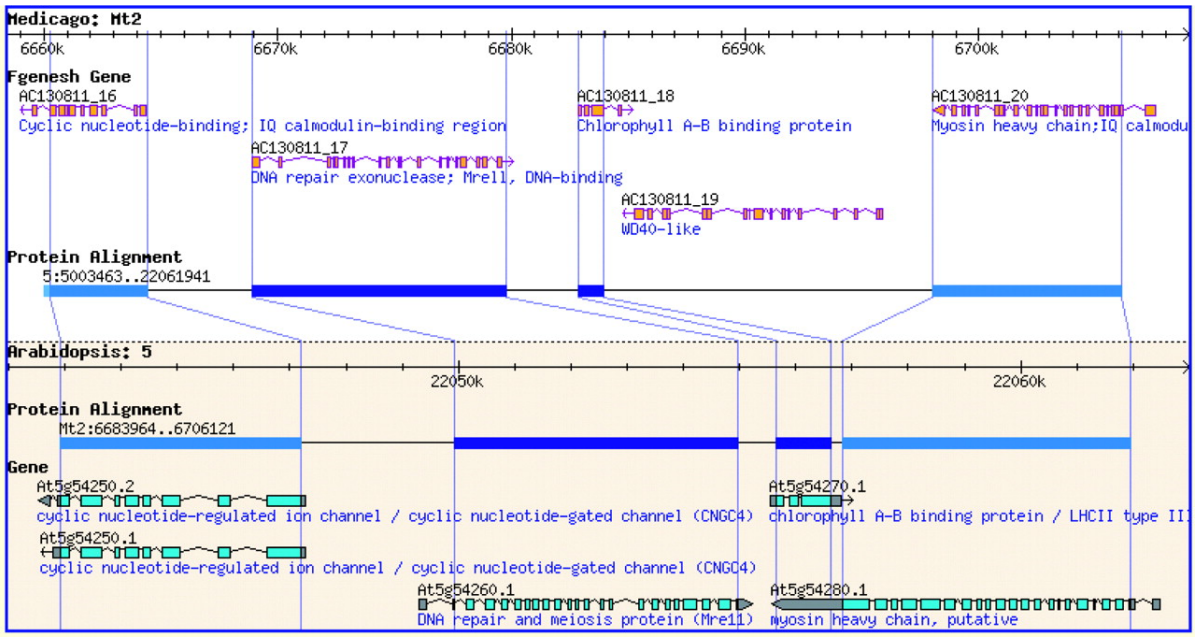


Figura 3.11: Comparação entre dois genomas pelo SynBrowse.

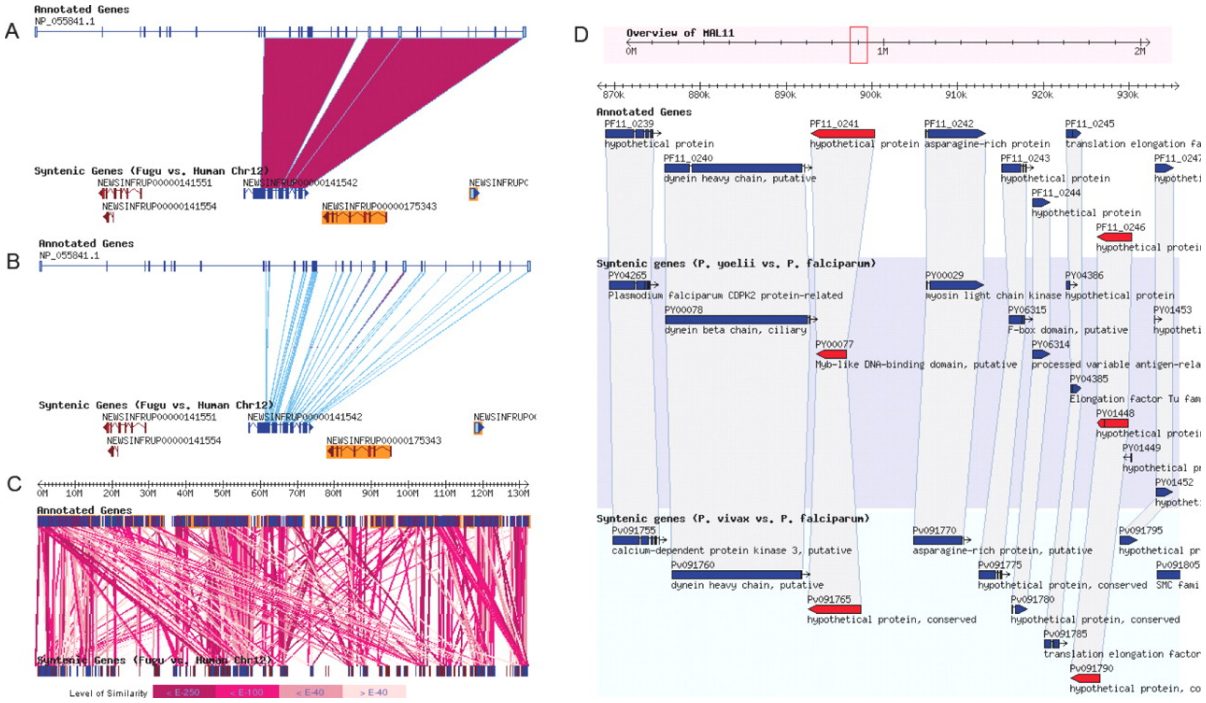


Figura 3.12: Comparação entre três genomas pelo SynView, com o genoma de referência ao topo.

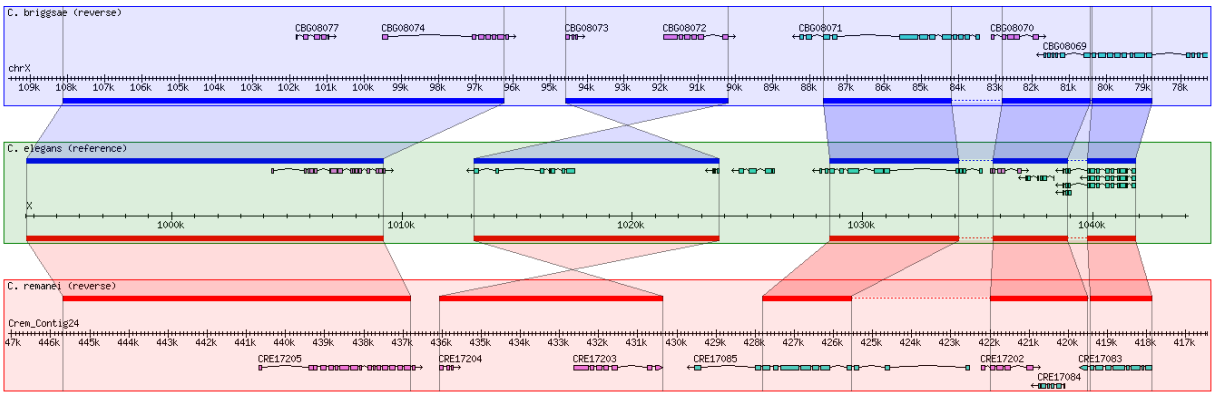


Figura 3.13: Exemplo da visualização disponibilizada pelo GBrowse_syn, mostrando a comparação entre três genomas.

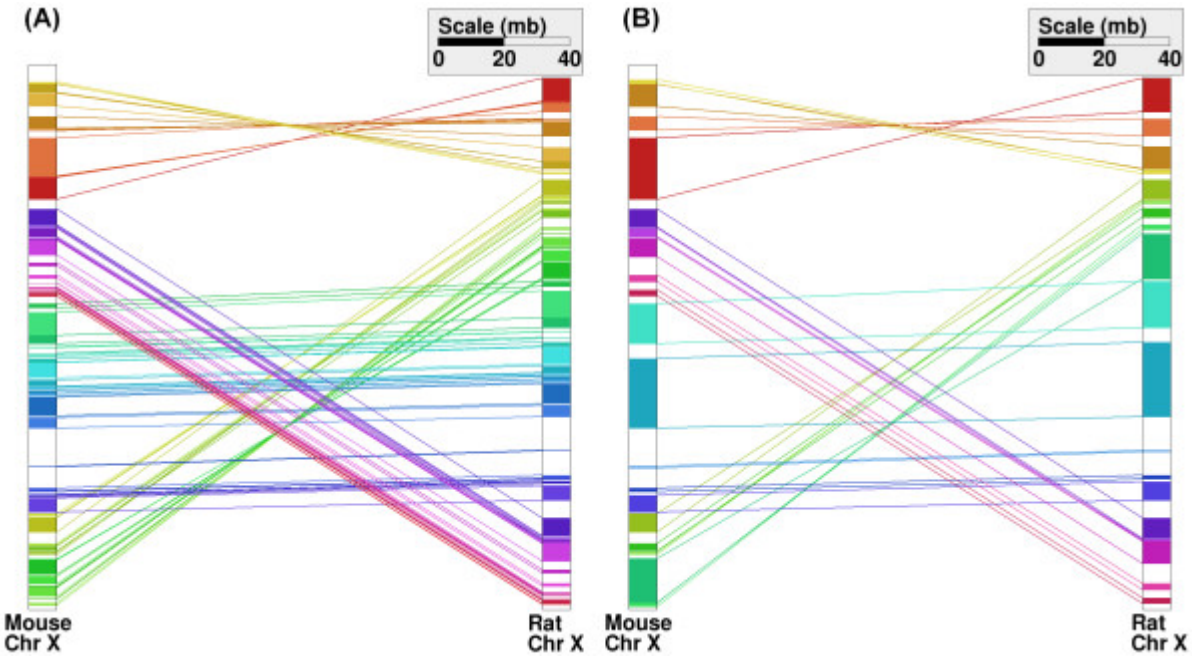


Figura 3.14: Comparação de cromossomos pelo Cinteny.

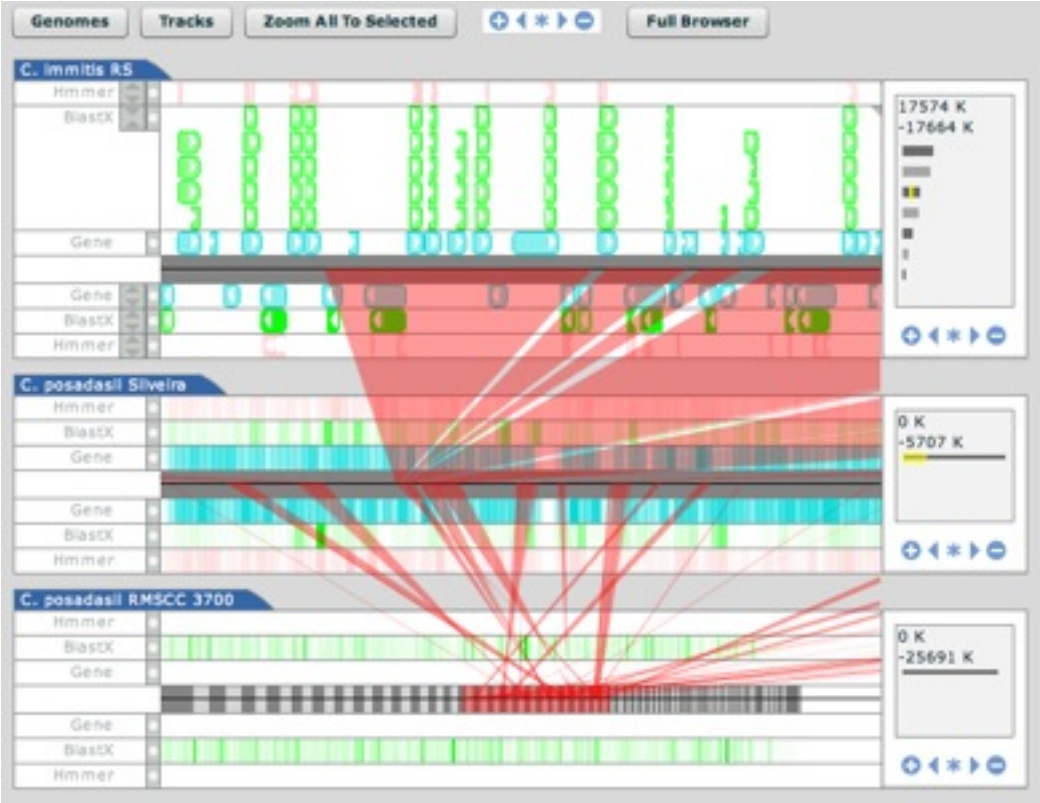


Figura 3.15: Módulo de visualização *Stack Map* do MEDEA, mostrando a comparação entre três genomas.

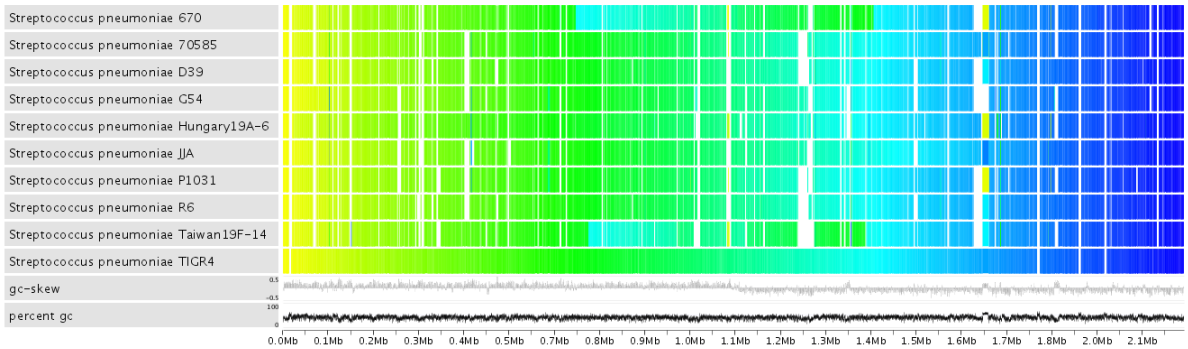


Figura 3.16: Visualização de gradiente de sintenia disponibilizada pelo Sybil.

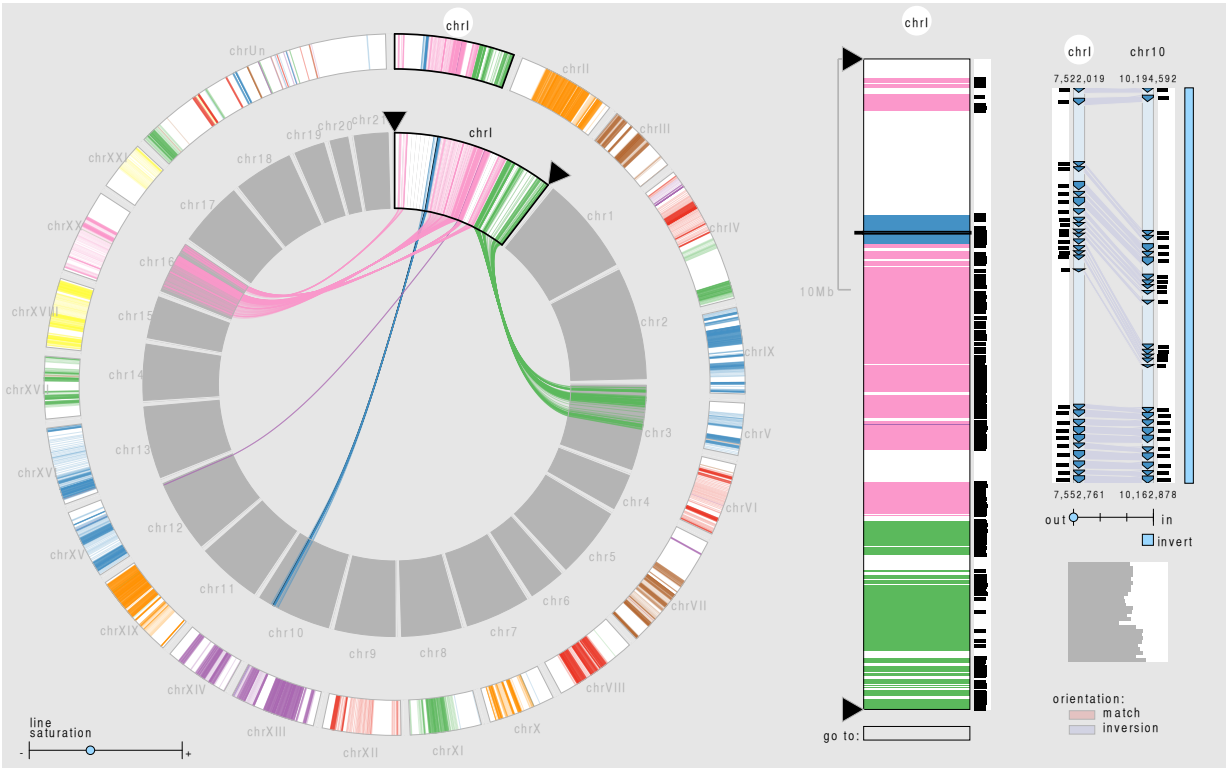


Figura 3.17: Os três níveis de visualização disponibilizados pelo MizBee.

Tabela 3.1: Ferramentas de visualização de sintenias, comparadas de acordo com os requisitos mais comuns e os tipos de gráficos. São comparados os requisitos definidos por Hunt et al. (2004) e os objetivos e tipos de gráficos propostos por Meyer et al. (2009).

Ferramentas	**Requisitos**								**Objetivos**														**Gráficos**					
	A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10	11	12	13	14	V1	V2	V3	V4	V5	V6
Apollo	•								•	•	•	•	•	•	•	•	•		•	•	•	•	•					
SyntenyView	•	•																	•	•	•	•						•
SyntenyVista	•	•	•	•	•	•	•			•	•	•	•	•	•	•	•		•	•	•	•	•					•
Mauve		•							•	•	•	•	•	•	•	•	•		•	•	•	•	•					
ACT		•							•	•	•	•	•	•	•	•	•		•	•	•	•	•					
SynBrowse	•	•							•	•	•	•	•	•	•	•	•		•			•	•					
SynView	•	•							•	•	•	•	•	•	•	•	•		•			•	•					
GBrowse_syn	•	•							•	•	•	•	•	•	•	•	•		•			•	•					
Cinteny	•	•	•				•		•	•	•	•	•	•	•	•	•		•	•	•	•	•				•	
MEDEA	•	•					•		•	•	•	•	•	•	•	•	•		•	•	•	•	•					
Sybil										•	•	•	•	•	•	•	•		•	•	•	•	•					
MizBee	•		•				•		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•				

Capítulo 4

Projeto Genoma Pb e sua Genômica Comparativa

Este capítulo descreve o fungo *Paracoccidioides brasiliensis* e seu projeto de sequenciamento genômico (Seção 4.1). Em seguida, na Seção 4.2 é apresentado o trabalho de genômica desenvolvido por de Carvalho (2010) e o método de visualização por esta proposto para a identificação de sintenias, o qual é o motivador do desenvolvimento do software Syntainia.

4.1 O Projeto Genoma Pb

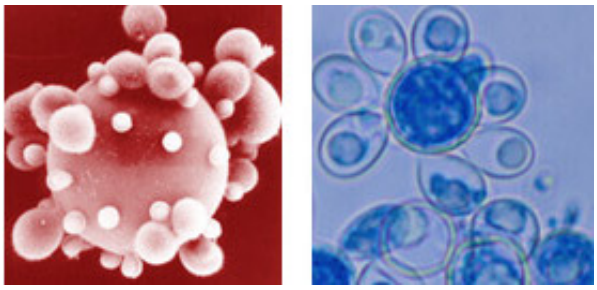
A biologia molecular de *P. brasiliensis* tem sido objeto de investigação, não apenas por sua importância clínica como mais importante micose sistêmica da América Latina, mas também como forma de compreender as micoses profundas como um todo e como modelo de infecções granulomatosas. Avanços recentes incluem a decodificação do transcriptoma diferencial das fases de levedura e micélio (Felipe et al., 2005a,b) e, mais recentemente, a elucidação do genoma cromossomal de três isolados do fungo. A seguir são apresentados o fungo *P. brasiliensis* e seu projeto de sequenciamento.

4.1.1 O fungo *P. brasiliensis*

O ascomiceto termodimórfico *Paracoccidioides brasiliensis* (Figura 4.1) pertence à classe Eurotiomycetes, subclasse Eurotiomycetidae, ordem Onygenales. A posição dele dentro dos Onygenales é algo controversa, não menos pelo fato de se tratar de um fungo mitospórico sem fase sexuada identificada até o momento. Embora originalmente classificado na família Onygenaceae, há um consenso filogenético emergente em posicioná-lo junto com *B. dermatitidis* e *H. capsulatum* na família Ajellomycetaceae, que incluiria a espécie-irmã *Lacazia loboi* (Paes, 2009; Teixeira et al., 2009).

P. brasiliensis é responsável pela paracoccidioidomicose (PCM). Antigamente chamada de blastomicose sul-americana, trata-se de uma micose sistêmica primária endêmica na região que se estende do México à Argentina, com maior incidência no Brasil (80% dos casos) e na região andina. O fungo foi primeiro isolado por

(a)



(b)

Figura 4.1: *P. brasiliensis*, em forma de micélio (a) e também em forma de levedura (b), adaptado de Broad Institute (2010b).

Adolfo Lutz em 1908. Estima-se que dez milhões de pessoas estejam infectadas, das quais cerca de duzentas mil desenvolverão a doença em algum momento de suas vidas (Felipe et al., 2005b). Embora esta doença seja potencialmente fatal e particularmente severa em crianças, pouco é ainda entendido sobre a biologia do *P. brasiliensis*. O hospedeiro humano adquire o fungo por inalação de esporos ou propágulos da forma miceliana presentes na natureza, que alcança os pulmões e se converte na forma de levedura. Recentemente, foi relatada a infecção de tatus em áreas endêmicas deste fungo, sendo portanto considerados hospedeiros silvestres naturais alternativos do homem. Contudo, não há registro de transmissão entre tatus e humanos (Paes, 2009).

In vitro, e provavelmente na natureza, o fungo dimórfico *P. brasiliensis* é encontrado como micélio ou esporo à temperatura ambiente ou na forma de levedura em temperaturas próximas a 37°C, sendo esta a forma predominante em tecidos de pacientes. Quando o *P. brasiliensis* infecta o hospedeiro, ocorre o processo de transição dimórfica, provavelmente disparada pela mudança de temperatura, ocorrendo a conversão da forma miceliana (ou esporo) para a forma de levedura. Estas observações sugerem fortemente que o processo dimórfico é um importante evento no estabelecimento da infecção, conforme observado para outros fungos patogênicos com *Candida albicans* e *Histoplasma capsulatum* (Felipe, 2010).

4.1.2 Projeto Genoma Funcional e Diferencial do *P. brasiliensis*

O projeto Genoma Funcional e Diferencial do *P. brasiliensis*, mais conhecido como Genoma Pb, baseia-se na identificação dos genes expressos em micélio e levedura, que potencialmente exerceriam funções relacionadas à adaptação do fungo ao hospedeiro, à manutenção do estado diferenciado, bem como com a virulência e/ou patogenicidade deste fungo, o qual sofre o processo de transição celular para infecção do hospedeiro humano. Esta proposta não se sobrepõe à do sequenciamento do seu genoma estrutural, ao contrário, complementa e avança no conhecimento da função dos genes, o que será certamento o passo seguinte na fase pós-genômica deste organismo. Além disto, o fato deste organismo possuir um genoma relativamente grande e complexo (30–60 Mbp) dificultaria uma proposta de genoma estrutural e traria como necessidade um maior número de grupos envolvidos em rede para alcançar este objetivo. Ao contrário, o genoma funcional e diferencial envolve o sequenciamento de ESTs de micélio e levedura e a identificação de genes diferenciais levando o projeto para uma escala viável de execução com um número menor de grupos em rede (Felipe, 2010).

Iniciado em 2001, o Projeto Genoma Pb tinha os seguintes objetivos (Felipe, 2010):

- Identificar e sequenciar de 10.000 a 15.000 ESTs de micélio e levedura do fungo *P. brasiliensis* (isolado Pb01).
- Mapear o genoma funcional diferencial do *P. brasiliensis*, por identificação dos genes expressos (ESTs), entre as formas de micélio e levedura do fungo.
- Caracterizar e efetuar a anotação funcional dos genes expressos.
- Implantar a infra-estrutura de Bioinformática na região Centro-Oeste.
- Qualificar técnicos e pesquisadores na área de Biotecnologia e Bioinformática na região Centro-Oeste.
- Melhorar a infra-estrutura dos laboratórios para projetos em rede, agregando grupos da região Centro-Oeste.

Finalmente, o Projeto Genoma Pb obteve os seguintes resultados (Felipe, 2010):

- 25.598 ESTs sequenciados.
- Total de 6.022 grupos obtidos e anotados:
- 2.655 contigs;
- 3.367 singlets.
- Primeiros resultados publicados em periódico em 2003 (Felipe et al., 2003).
- Inúmeros trabalhos de graduação, dissertações de mestrado e teses de doutorado em Biologia Molecular e Ciência da Computação.

- Consolidação do Laboratório de Bioinformática da UnB (Brígido et al., 2005), com a implementação de inúmeras ferramentas computacionais, e o desenvolvimento de novos projetos de sequenciamento (Coimbra et al., 2007).

Mais recentemente, os dados produzidos pelo Projeto Genoma Pb foram incorporados ao banco de dados ***Paracoccidioides brasiliensis*** **Database**, mantido pelo Broad Institute (2010a). Além do isolado Pb01, esse banco de dados conta com informações de mais dois isolados de *P. brasiliensis*: Pb18 e Pb03. Os isolados Pb01 e Pb18 foram alvos de estudos de expressão diferencial (Felipe et al., 2005a, 2003; Nunes et al., 2005), e, no caso de Pb01, atingiu-se uma cobertura de mais de 80% dos mRNAs produzidos pelo fungo nas fases de micélio e levedura (Felipe et al., 2005a); Pb18, por sua vez, é padrão para estudos de virulência e imunidade (Nunes et al., 2005). Pb03, apesar de menos estudado, foi incluído no projeto de sequenciamento do genoma cromossomal e mitocondrial de *P. brasiliensis*, por representar um terceiro grupo de espécie filogenética (Matute et al., 2006).

Estudos de genômica comparativa recentes sugerem que o grau de especialização do isolado Pb01 faz com que este caracterize-se como uma espécie diferente dos outros dois isolados (Teixeira et al., 2009). Nesse sentido, foi proposta a criação do gênero *Paracoccidioides* e que o isolado Pb01 seja denominado como *Paracoccidioides lutzii*, em homenagem ao pesquisador brasileiro Adolfo Lutz.

4.2 Genômica Comparativa e um Novo Método de Visualização

A conclusão do trabalho de sequenciamento de *P. brasiliensis*, com a anotação genômica dos isolados Pb01, Pb03 e Pb18, abriu caminho para pesquisas de genômica comparativa com outros fungos patogênicos e não-patogênicos bem conhecidos. Tais estudos tem sido direcionados no entendimento da filogenia do fungo (Teixeira et al., 2009) e também no estudo de como identificar características genômicas que auxiliem no tratamento da PCM, com base em características semelhantes em outros fungos (de Carvalho, 2010), entre outros trabalhos.

O trabalho desenvolvido por de Carvalho (2010) tem como objetivo geral investigar a localização dos genes potencialmente envolvidos nos processos de virulência ou patogenicidade, e genes essenciais, os quais possivelmente estão relacionados ao processo de infecção de *P. brasiliensis*. Os genes escolhidos são aqueles que já foram descritos em trabalhos relacionados com MDR ("Multi-Drug Resistance"), estresse oxidativo (RNI e ROS), potenciais alvos para drogas, transdução de sinal, choque térmico, interação patógeno-hospedeiro, virulência e genes essenciais. Esta investigação está sendo realizada através de análise comparativa entre os genomas estruturais de Pb01 e *A. fumigatus*, *A. nidulans*, *C. immitis*, *H. capsulatum*, *C. albicans*, *N. crassa*, *S. cerevisiae*, Pb03 e Pb18. Esse trabalho de genômica comparativa tem ainda os seguintes objetivos específicos:

- Seleção dos genes ou transcritos relacionados às categorias mencionadas acima para a análise de sintenia entre fungos patogênicos (*H. capsulatum*, *C. immi-*

tis, *A. fumigatus*, *C. albicans* e *P. brasiliensis* — isolados Pb01, Pb03, Pb18) e não-patogênicos (*A. nidulans*, *N. crassa* e *S. cerevisiae*);

- Localização dos genes ou transcritos selecionados nos fungos patogênicos e não-patogênicos;
- Análise de sintenia comparativa dos genes ou transcritos selecionados nos isolados Pb01 (*P. lutzii*), Pb03 e Pb18, utilizando o software Syntainia, desenvolvido em colaboração durante a realização deste trabalho;
- Análise de sintenia comparativa dos genes ou transcritos selecionados nos fungos patogênicos e não-patogênicos;
- Análise de sintenias que podem estar relacionadas com a patogenicidade ou virulência entre os fungos patogênicos e não-patogênicos selecionados;
- Proposição de uma estrutura organizacional de grupos de genes ou transcritos relacionadas com a patogenicidade ou virulência entre os fungos patogênicos e não-patogênicos analisados.

Nesse sentido, o trabalho proposto por de Carvalho (2010) demandava a utilização de um mecanismo de visualização capaz de evidenciar as relações entre os genes ou transcritos de *P. lutzii* e os genes de outros fungos. Era necessário que a visualização de um volume tão grande de dados fosse feita de forma bastante clara e com o foco em como os genes ou transcritos estão agrupados e se conservam entre múltiplos genomas, visto que dentre os objetivos do trabalho está a análise de sintenias entre fungos. Assim, foi possível observar que nenhuma das ferramentas apresentadas no Capítulo 5 atendia satisfatoriamente a esses requisitos, o que motivou o desenvolvimento de um novo método de visualização dos genes.

4.2.1 Método de Visualização dos Genes

A seguir é descrito o trabalho desenvolvido por de Carvalho (2010), que veio a se consolidar num método para visualização de genes ou transcritos.

Escolha de genes ou transcritos

A análise do transcriptoma do *P. brasiliensis*, linhagem Pb01, encontrou 6.022 PbA-ESTs ("Pb Assembled Expressed Sequence Tags") e a anotação destes permitiu identificar a que genes correspondiam e, consequentemente, suas categorias funcionais (Felipe et al., 2005a,b). O critério para a escolha foi a de que os genes pertencessem a categorias consideradas como potenciais alvos de drogas e de envolvimento com virulência e patogenicidade.

Obtenção das sequências genômicas

As sequências das três linhagens de *P. brasiliensis* foram obtidas do banco de dados do Broad Institute[1]. As sequências genômicas dos fungos patogênicos *C. im-*

[1]Sequências disponíveis para download em `http://www.broadinstitute.org/annotation/genome/paracoccidioides_brasiliensis/MultiDownloads.html`.

mitis, *H. capsulatum*, *A. fumigatus* e *C. albicans* e dos fungos não patogênicos *A. nidulans*, *N. crassa* e *S. cerevisiae* foram obtidas do banco de dados do NCBI. As sequências obtidas dos bancos de dados do Broad Institute e do NCBI contém, em sua anotação, informação sobre a quais cromossomos, *supercontigs* ou *scaffolds* cada sequência pertence. De fato, as sequências rotuladas de cada banco de dados correspondem a cromossomos, *supercontigs* ou *scaffolds* inteiros, nos quais podem estar presentes diversos genes.

Busca por ortólogos dos genes selecionados nos genomas

A comparação foi feita com o uso da ferramenta tBLASTx que compõe o pacote de programas BLAST. Esta ferramenta primeiro traduziu, nas seis fases de leitura, tanto as sequências dos PbAESTs, cujos ortólogos se queria encontrar, e que foram chamados individualmente de *Query*, quanto os genomas, tratados como banco de dados onde foi feita a pesquisa, e que os tornaram *Subjects*. O próximo passo foi a comparação entre as sequências de resíduos de aminoácidos dos *Queries* e dos *Subjects*. Cada arquivo gerado correspondia a um resultado da comparação entre os genes selecionados e um dos genomas. Assim foram obtidos dez arquivos distintos para serem usados na próxima etapa.

Elaboração do gráfico

A identificação de quais genes que apresentariam a mesma organização nos genomas selecionados de fungos patogênicos e não patogênicos para humanos poderia ser feita com os dados provindos dos resultados obtidos do BLAST. Mas era preciso visualizar essa organização. Então, fazendo uso da informação referente à identificação da posição do primeiro e último nucleotídeos da região onde melhor houve alinhamento com um dos genes, ou transcritos, usados foi possível posicionar cada um destes nos genomas já mencionados. Como a intenção era de apenas visualizar como os genes ou transcritos estavam organizados nos genomas, não houve a necessidade dos fragmentos serem representados respeitando seus tamanhos. Assim, também foram desconsideradas as sequências entre os ortólogos dos genes ou transcritos analisados.

O gráfico idealizado é formado basicamente dos números de PbAESTs ordenados verticalmente como retângulos coloridos. A coloração se dá de acordo com a categoria funcional a qual pertencem as proteínas codificadas pelos ortólogos dos genes ou transcritos de Pb01. Esses retângulos são agrupados, formando colunas que representavam os fragmentos genômicos (cromossomos, *supercontigs* ou *scaffolds*), devidamente rotulados, nos quais seus ortólogos foram encontrados. A ordenação delas, como já foi dito, é baseada na localização da região que mais apresentou similaridade, no caso, o primeiro melhor resultado considerado pelo programa tBLASTx. Cada genoma então é nomeado e disposto verticalmente. Finalmente, os ortólogos são ligados por linhas entre cada um dos genomas comparados. A Figura 4.2 ilustra parte de um gráfico, mostrando seis genomas, com os genes ou transcritos de Pb01 mapeados e agrupados de acordo com os cromossomos, *supercontigs* ou *scaffolds* aos quais pertencem os seus ortólogos em cada um dos seis genomas.

Resumo do método

Resumidamente, o método proposto é composto das seguintes etapas:

1. Seleção dos genomas com os quais as sequências em estudo serão comparadas. As sequências desses genomas devem estar agrupadas de alguma forma (por exemplo, em cromossomos, *supercontigs* e *scaffolds*).
2. Execução de BLAST — mais especificamente, da variante tBLASTx — de todas as sequências em estudo contra as sequências dos genomas selecionados.
3. Análise dos arquivos de resultado de cada BLAST, selecionando sempre o melhor *hit* de cada comparação. O melhor *hit* é considerado como o ortólogo do sequência em estudo no genoma comparado.
4. Agrupamento dos melhores *hits* de acordo com a anotação de cada genoma (por exemplo, os *hits* serão agrupados em cromossomos, supercontigs e scaffolds).
5. Ordenação dos *hits* dentro de cada grupo de acordo com a posição do alinhamento fornecido pelo BLAST.
6. Rotulação dos *hits* de acordo com uma regra que seja conveniente para o pesquisador e que torne clara a visualização no gráfico.
7. Geração de gráfico no qual os genomas são mostrados lado a lado, verticalmente, com os *hits* agrupados e ligados por linhas com os seus respectivos ortólogos no genoma vizinho.

Em essência, o método de visualização descrito, além de fornecer uma comparação entre múltiplos genomas, também faz o mapeamento de um genoma incompleto em genomas que estejam completos ou em fase de montagem. No caso, o trabalho realizado por de Carvalho (2010) mapeia as sequências de Pb01 obtidas do Projeto Genoma Pb nas sequências de Pb03, Pb18 e de outros fungos similares.

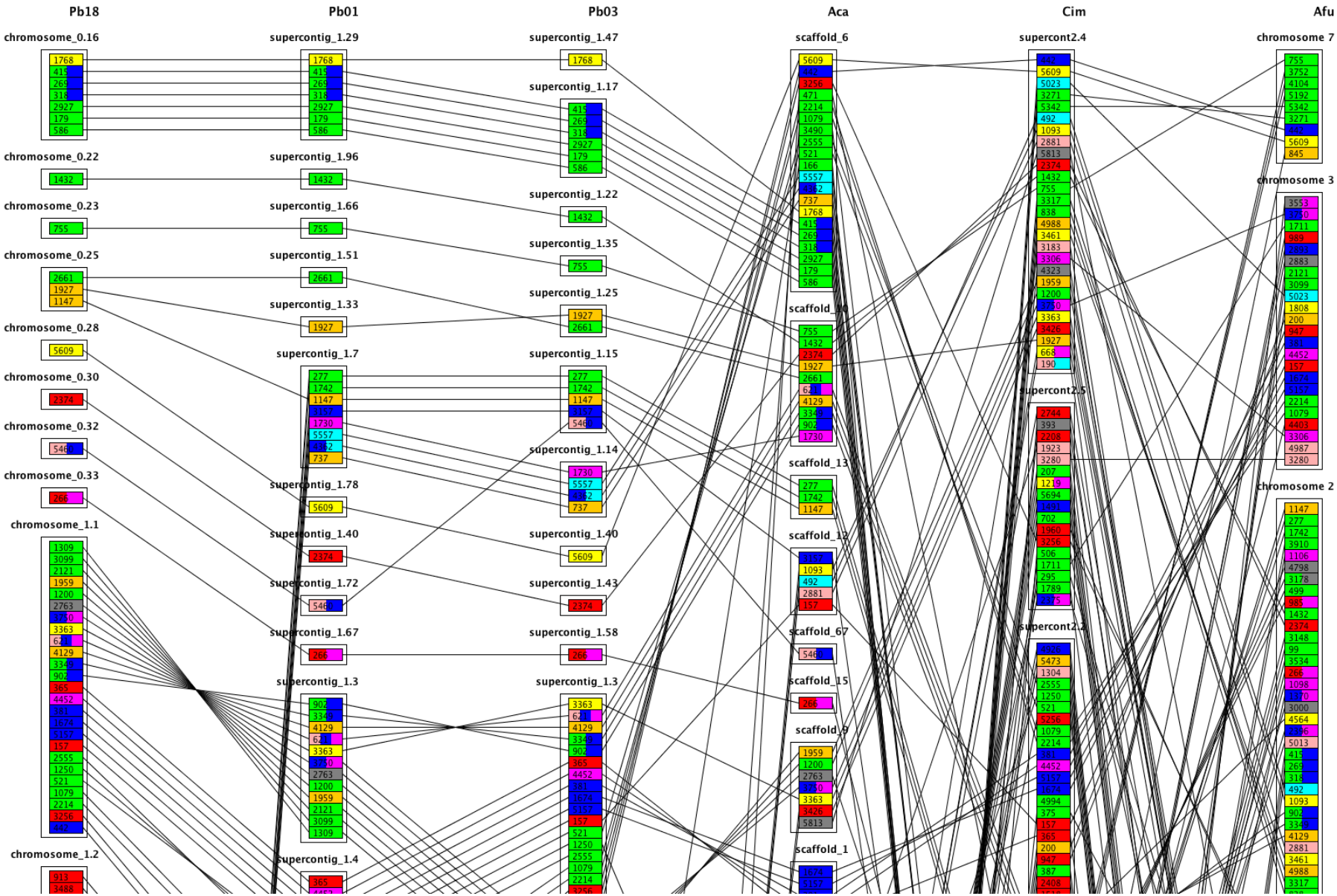


Figura 4.2: Exemplo da visualização de múltiplos genomas por grupos de genes.

Capítulo 5

Proposta e Implementação da Ferramenta

Os trabalhos de análise do genoma do *P. brasiliensis* culminaram na elaboração de uma nova forma de visualizar genomas, conforme apresentado no Capítulo 4, o que motivou a implementação de uma ferramenta computacional. Este capítulo apresenta essa ferramenta, denominada *Syntainia*, seus requisitos, algoritmos e arquitetura.

Na Seção 5.1 são apresentados os elementos que orientaram o desenvolvimento do Syntainia. Em seguida, a Seção 5.2 enumera os requisitos que o software implementa. Na Seção 5.3 são apresentadas as principais características do software implementado e suas funcionalidades. A Seção 5.4 descreve dois algoritmos de otimização para a visualização dos genomas. Finalmente, a Seção 5.5 descreve a arquitetura do Syntainia, assim como detalhes de sua implementação.

5.1 Visão Geral

O trabalho de genômica comparativa realizado por de Carvalho (2010) obteve, como um dos resultados, um novo método de visualização de comparações entre múltiplos genomas, voltado para a identificação de sintenias. Assim, para apoiar a genômica comparativa executada por de Carvalho (2010) foi necessário o desenvolvimento de um software que implementasse esse novo método. Essa implementação foi consolidada com o elaboração deste trabalho, realizado em conjunto com de Carvalho (2010), resultando no desenvolvimento do software denominado *Syntainia*.

O nome *Syntainia* é baseado na origem etimológica da palavra *sintenia*. A palavra *sintenia* deriva do termo *synteny* da língua inglesa, que trata-se de um neologismo com o significado de “on the same ribbon” (“na mesma fita”). Esse neologismo é baseado nas palavras gregas que dão o significado desejado: o prefixo *syn* (no mesmo, juntos) e o sufixo *tainia* (faixa, fita), que foi adaptado para *teny*, visando facilitar a pronúncia em língua inglesa. Dessa forma, *Syntainia* resgata a origem do termo *sintenia* com a composição das palavras gregas que inspiraram a criação dessa nova palavra.

Syntainia tem sido desenvolvido como um software livre. Está sendo utilizada a infraestrutura disponibilizada pelo portal SourceForge.net para a manutenção e

distribuição do software (Geeknet, Inc., 2010). Foi criado um projeto denominado Syntainia[1], a partir do qual é possível navegar pelo código-fonte, obter informações sobre o projeto, relatar problemas e sugerir modificações. Tem sido utilizada a ferramenta de controle de versão Subversion, disponibilizada pelo portal. E seguindo o princípio do "libere cedo, libere frequentemente", comumente adotado em projetos de software livre (Raymond, 2000), já há uma versão disponível para download do Syntainia na página do projeto no SourceForge.net. Syntainia é distribuído sob a licença GNU General Public License (GPL), versão 2 (FSF, 2010).

Além de implementar o método de visualização proposto por de Carvalho (2010), Syntainia também o aprimora por meio da aplicação de algoritmos que visam melhorar a organização do gráfico gerado, com o objetivo de tornar mais clara a visualização dos genomas comparados. Esses algoritmos serão detalhados na Seção 5.4. Outra característica adicionada ao Syntainia é a possibilidade de o usuário interagir com o gráfico. Isso significa que o pesquisador é capaz de rearranjar genomas e grupos de genes, de modo a obter a visualização que lhe for mais conveniente. Tais características do Syntainia serão melhor exploradas na Seção 5.3.

Ao implementar o método proposto por de Carvalho (2010), baseado sobretudo na aplicação sucessiva do BLAST, Syntainia também faz, em essência, o mapeamento de um genoma incompleto em genomas bem estudados e que estejam mais completos. Essa característica faz com que Syntainia possa ser utilizado em outras tarefas de Bioinformática, como o mapeamento de genomas, além do auxílio na identificação de sintenias.

5.2 Requisitos

O desenvolvimento do Syntainia buscou basicamente atender às necessidades do trabalho de genômica comparativa realizado por de Carvalho (2010). Além disso, foram considerados aspectos de projetos anteriormente desenvolvidos no Laboratório de Bioinformática da UnB (Coimbra et al., 2007), com vistas à evolução do software. Nesse sentido, os principais requisitos que orientaram o desenvolvimento do Syntainia foram:

- **Visualização dos genomas como grupos de genes**: a essência do método de visualização proposto por de Carvalho (2010) deve ser implementada.
- **Arranjo automático dos dados**: devem ser implementadas técnicas de arranjo dos dados, que tem como objetivo fornecer a melhor visualização possível para a identificação de sintenias.
- **Interatividade com o gráfico**: a interface gráfica deve favorecer o rearranjo dos dados e alteração de formas e cores, facilitando a identificação de sintenias.
- **Linguagem Java**: o software deve ser escrito em Java, visando o aproveitamento de código de outros projetos e a integração com trabalhos futuros.

[1]Página do projeto: `http://sourceforge.net/projects/syntainia`.

- **Aplicativo *desktop***: visando a melhor usabilidade, o software deve ser implementado como um aplicativo *desktop*, projetado para a máxima integração possível com o sistema operacional.
- **Exportar o gráfico para diversos formatos**: o gráfico deve ser exportado para diversos formatos, tais como SVG, PNG, BMP e JPEG.
- **Seleção do tipo de entrada**: o software deve ser capaz de processar arquivos de resultado do BLAST previamente gerados ou receber arquivos de sequências em formato FASTA e executar o BLAST.
- **Executar outros programas de comparação**: deve ser possível utilizar outros softwares de comparação de sequências, além do BLAST.
- **Flexibilidade no agrupamento dos genes**: o critério de agrupamento dos genes deve ser flexível, permitindo a composição de cromossomos, *supercontigs*, *scaffolds* ou o que for mais conveniente para o pesquisador.
- **Categorização dos genes**: o pesquisador deve poder fornecer dados sobre categorias ou alguma forma de caracterização de cada gene.
- **Rotulação simples**: deve ser possível fornecer nomes curtos (rótulos) para os genes, a fim de simplificar seu desenho no gráfico.

Como o Syntainia foi projetado como uma ferramenta de visualização de comparações entre genomas, objetivando a identificação de sintenias, é necessário considerar os requisitos comumente considerados na elaboração de outras ferramentas. Assim, considerando os requisitos definidos por Hunt et al. (2004), que tratam essencialmente da visualização disponibilizada pela ferramenta, Syntainia implementa as seguintes características:

A) **Detalhes sob demanda**: o usuário deve ser capaz de selecionar os genomas e grupos de genes (cromossomos, *supercontigs* ou *scaffolds*) que deseja visualizar.

B) ***Zoom***: não implementado.

C) **Rotulação eficiente**: os rótulos dos genes devem ter nomes curtos.

D) **Movimentar um cromossomo ao longo do seu eixo**: não implementado.

E) **Aplicar escala ao cromossomo**: exibir os genes em grupos, com retângulos do mesmo tamanho, ignorando o tamanho e a distância entre eles, o que caracteriza o *cartoon scaling*.

F) **Inversão do cromossomo**: não implementado.

G) **Filtragem**: onde há muitos dados para serem exibidos, deve haver a possibilidade de filtrar os dados de acordo com certas regras.

H) **Coloração**: o software deve usar um esquema de coloração por padrão para diferenciar genes e as linhas que os ligam a seus ortólogos, mas deve permitir que o usuário modifique as cores.

Outro conjunto de características que devem ser consideradas durante a implementação de ferramentas de visualização é relativo aos aspectos que o pesquisador tenta observar ao realizar um estudo sobre sintenias. Tais características foram levantadas por Meyer et al. (2009), dos quais muitos podem ser observados pelo Syntainia:

1. Quais cromossomos compartilham blocos de genes conservados: **implementado**.
2. Para um cromossomo, com quantos outros cromossomos ele compartilha blocos de genes conservados: **implementado**.
3. Qual a densidade de cobertura do genoma e onde estão os *gaps*: **não implementado**.
4. Onde estão os blocos de genes, se estariam em torno de uma posição específica no cromossomo: **implementado**.
5. Quais são os tamanhos e localizações de outras características genômicas próximas a um bloco de genes: **implementado**.
6. Quão grandes são os blocos de genes: **implementado**.
7. Se blocos de genes vizinhos se conservam no mesmo cromossomo e/ou preservam seu posicionamento relativo: **implementado**.
8. Se a orientação dos pares de blocos de genes é preservada ou invertida: **implementado**.
9. Se a orientação é preservada para blocos de genes vizinhos: **implementado**.
10. Se as pontuações de similaridades são iguais com respeito a blocos de genes vizinhos: **não implementado**.
11. Se os genes pareados dentro de um bloco são contíguos: **implementado**.
12. Quão grande é um gene em relação a outros genes dentro de um bloco: **não implementado**.
13. Quais são os tamanhos, localizações e nomes dos genes dentro de um bloco: **implementado**.
14. Quais são as diferenças entre nucleotídeos individuais e pares de genes: **não implementado**.

No Capítulo 6 é feita uma discussão acerca dos requisitos implementados pelo Syntainia e as principais características de outras ferramentas. Na Tabela 6.1 é apresentada uma comparação do Syntainia com as ferramentas discutidas na Seção 3.3, de acordo com os requisitos e objetivos anteriormente citados.

5.3 Características do Syntainia

Nesta seção são apresentadas as principais características do Syntainia, tais como aspectos do projeto da sua interface gráfica com o usuário, o modo de visualização da comparação entre os genomas e as funcionalidades que estão à disposição do pesquisador. Tais características são o resultado do levantamento de requisitos descrito na Seção 5.2.

5.3.1 Interface com o Usuário

Syntainia é um aplicativo *desktop*, baseado em Java Swing (Oracle Corporation, 2010b), cuja interface gráfica foi projetada para se adequar ao *look and feel*[2] dos sistemas operacionais mais utilizados — Windows, Mac OS X, Linux e outros sistemas UNIX que utilizam GNOME. Em geral, a maioria dos programas que se baseiam em Java Swing utiliza o *look and feel* padrão, que se apresenta de maneira uniforme entre os diversos sistemas operacionais. Isso significa que um aplicativo que usa o *look and feel* padrão não se parece com nada que o usuário está acostumado no sistema operacional que ele utiliza, tanto sob o aspecto meramente visual, quanto em relação à localização de itens de menu e até mesmo teclas de atalho (Faaborg, 2007). A Figura 5.1 apresenta como seria a janela principal do Syntainia utilizando o *look and feel* padrão.

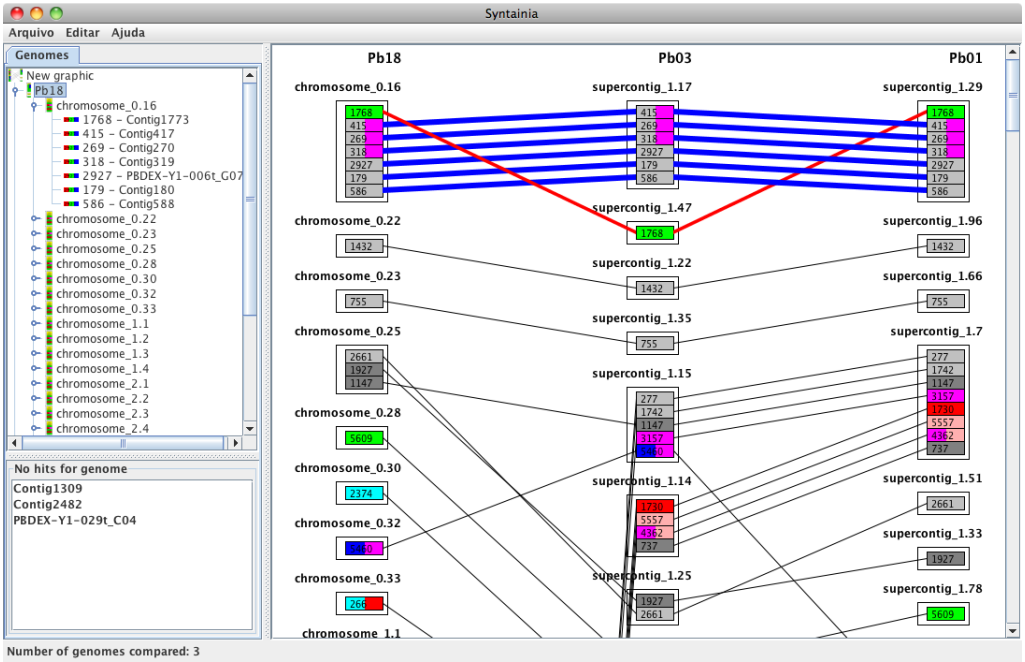


Figura 5.1: Usando o *look and feel* padrão na janela principal do Syntainia. Apesar de estar sendo executado no Mac OS X, a aplicação não se parece em nada com um software nativo para essa plataforma.

[2]Termo utilizado em relação a interface gráfica do usuário e compreende os aspectos da sua concepção, incluindo elementos como cores, formas, disposição e tipos de caracteres (o "Look"), bem como o comportamento de elementos dinâmicos, tais como botões, caixas e menus (o "Feel").

Nesse sentido, foram seguidas as recomendações de projeto de interface dos principais sistemas operacionais, descritas na Seção 5.5. Assim, o Syntainia ajusta sua interface gráfica ao sistema operacional no qual é executado. Além disso, a disposição de itens de menu e botões é modificada, de modo a se ajustar às expectativas do usuário, considerando a experiência que ele já possui com o sistema operacional que está acostumado a utilizar. Dessa forma, se um aplicativo se parece visualmente com programas nativos do sistema operacional e se comporta como tal, usar esse aplicativo é uma atividade mais natural e o aprendizado sobre como utilizá-lo torna-se mais simples (Faaborg, 2007). Na Figura 5.2 são apresentadas telas do Syntainia conforme são exibidas no Windows, Mac OS X, Linux e outros sistemas UNIX que utilizam GNOME.

A interface gráfica do Syntainia é composta essencialmente por uma janela principal, na qual o gráfico das comparações entre os genomas é exibido. Outro elemento importante da interface gráfica do aplicativo é o assistente para geração de um novo gráfico.

Janela Principal

A janela principal do Syntainia é apresentada na Figura 5.3. O elemento que domina a janela principal da aplicação é o painel de exibição do gráfico com a comparação entre os múltiplos genomas submetidos pelo usuário. É possível navegar pelo gráfico, porém, ainda não é possível fazer *zoom* na área de exibição ou mover os objetos diretamente sobre ela. Na Subseção 5.3.2 são apresentadas todas as características do gráfico gerado pelo Syntainia.

Além de exibir o gráfico, a janela principal contém um painel à esquerda, no qual o usuário tem à disposição uma visão de árvore de todos os dados disponíveis. A árvore reproduz, em sua hierarquia, a mesma estrutura na qual os dados são organizados no gráfico, ou seja, genomas compostos por grupos de genes. A partir da árvore é possível selecionar os elementos que serão exibidos no gráfico, além de personalizar a exibição disponível, alterando cores e largura de linhas. Essas funcionalidades estão disponíveis a partir de um menu contextual sobre os nós da árvore. Encontra-se também no painel à esquerda uma lista dos genes não mapeados, de acordo com o genoma selecionado na árvore.

Na parte inferior da janela principal encontra-se uma barra na qual são exibidas mensagens sobre as tarefas executadas pelo Syntainia. Em especial, após a geração de um novo gráfico é exibida uma mensagem que sinaliza quantos genomas foram comparados.

Assistente

Através do assistente, o usuário é guiado através dos passos necessários para a criação de um novo gráfico de comparação entre múltiplos genomas, tais como seleção dos arquivos de entrada de dados. Uma das telas do assistente é ilustrada pela Figura 5.4. A Subseção 5.3.3 lista todas as etapas do assistente.

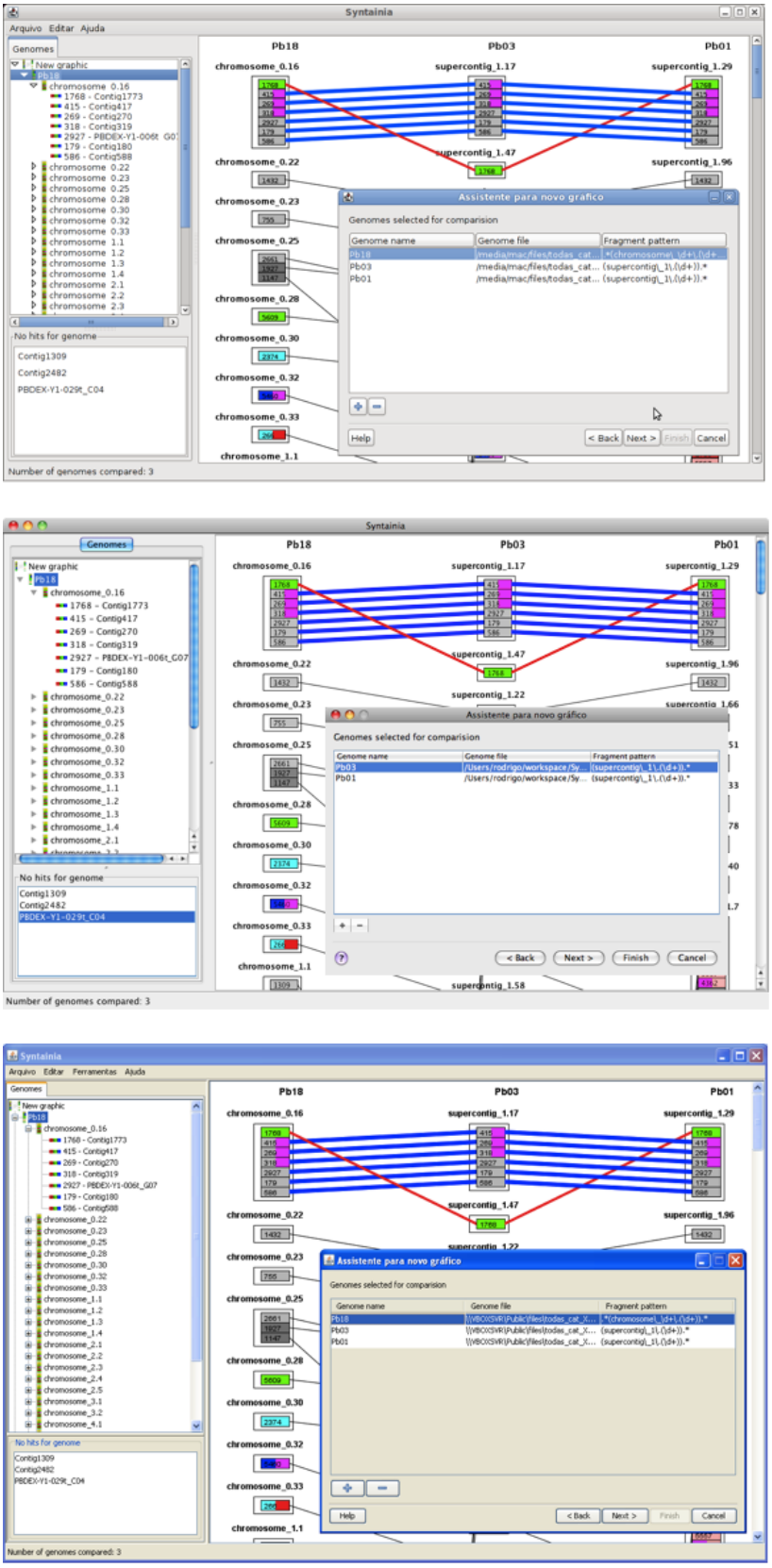


Figura 5.2: Comparação das telas do Syntainia em diversos *look and feels*. Em (a) é mostrada a aparência num sistema GNOME; em (b) pode ser vista a aparência no Mac OS X; e em (c) é apresentada a interface no Windows XP.

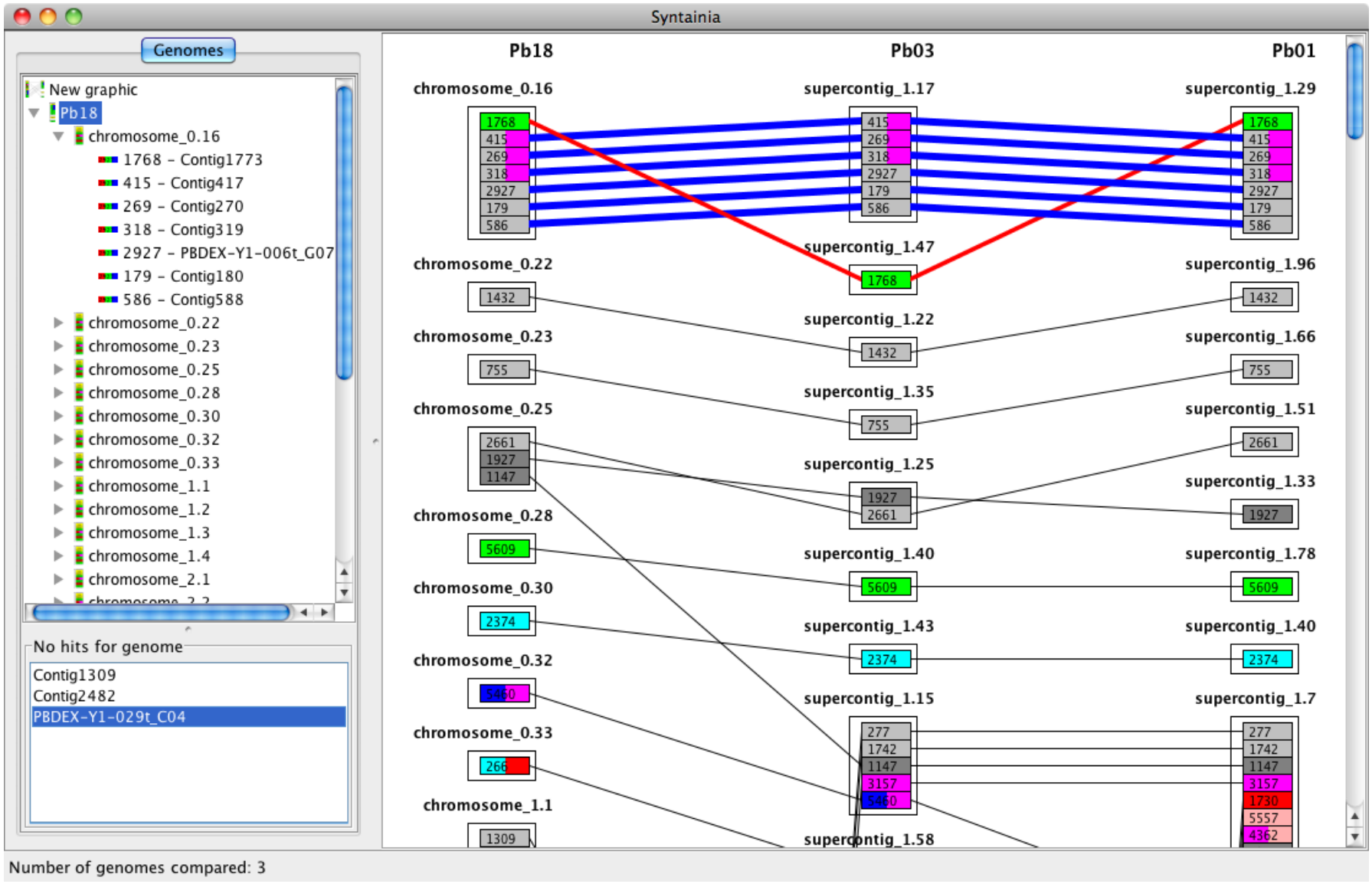


Figura 5.3: Janela principal do Syntainia, para visualização e interação com o gráfico.

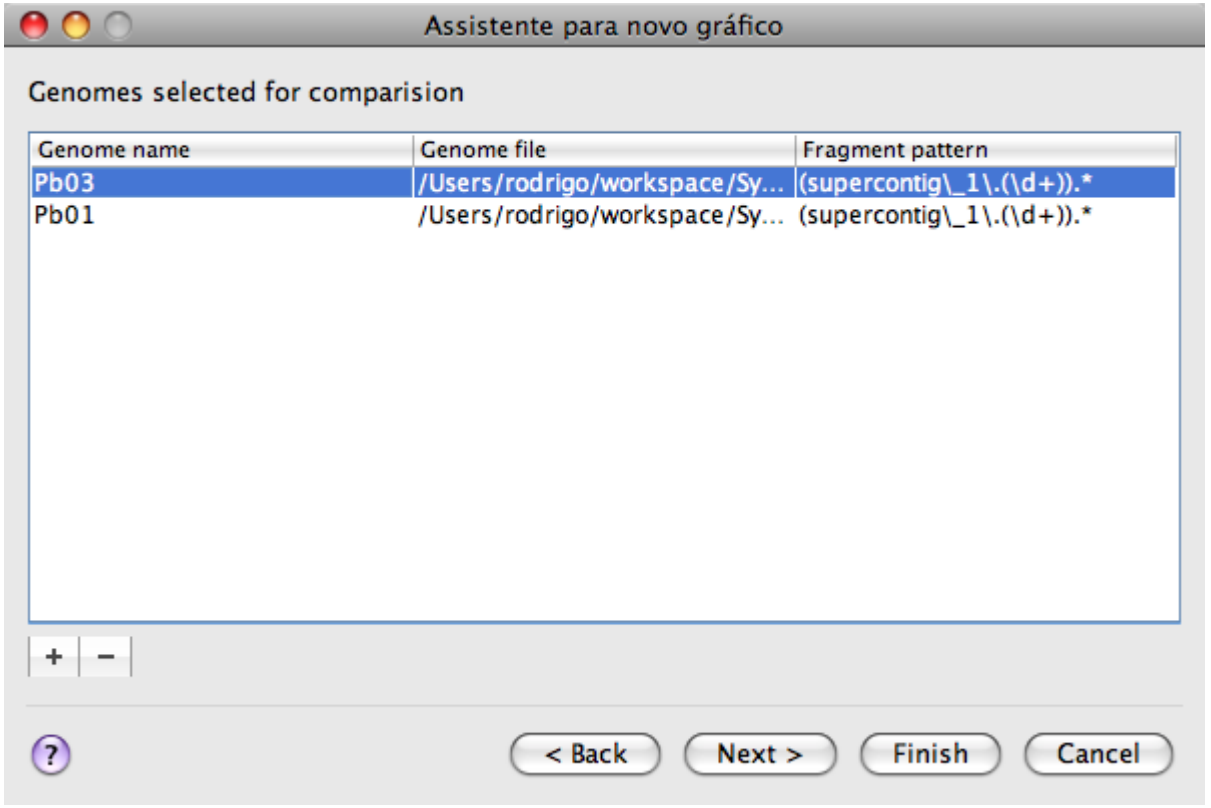


Figura 5.4: Tela do assistente para geração do gráfico.

5.3.2 Modo de Visualização

Syntainia exibe um genoma como uma lista de grupos de genes, orientados verticalmente, tal como proposto no Capítulo 4 (de Carvalho, 2010). Como dito anteriormente, esses grupos de genes podem ser cromossomos, *supercontigs*, *scaffolds* ou qualquer outra forma de organização conveniente para o usuário. Os grupos são organizados de forma automática dentro do genoma, utilizando um dos dois algoritmos descritos na Seção 5.4, de modo que o gráfico tenha o menor número possível de linhas que se cruzam. Dentro do grupo, os genes são ordenados de acordo com o seu posicionamento estrutural no genoma ou, mais especificamente, no grupo.

Cada gene é colorido de acordo com suas categorias genômicas, conforme informado pelo arquivo de dicionário de dados — arquivo tabulado que associa um gene a um nome curto e a uma lista de categorias genômicas. Quando um gene pertence a mais de uma categoria, então são coloridos retângulos correspondentes a cada categoria dentro do retângulo que representa o gene.

Os nomes curtos provenientes do dicionário de dados são utilizados como rótulos dos genes. Caso não sejam encontrados nomes curtos, os nome originais dos genes são utilizados. Além disso, cada gene é ligado por uma linha a seu respectivo ortólogo no genoma imediatamente à esquerda. Os rótulos de cada grupo são o resultado da indicação feita pela expressão regular que os determina em cada genoma. Finalmente, os rótulos de cada genoma são fornecidos pelo usuário através do assistente, durante a seleção dos arquivos de entrada.

Considerando os modos de visualização possíveis para a exibição da comparação entre múltiplos genomas apresentada por Meyer et al. (2009), Syntainia disponibiliza um gráfico do tipo **discreto intercalado** (ver Figura 3.5), uma vez que a visualização apresentada é organizada por grupos de genes que podem ser arbitrariamente dispostos a fim de tornar mais claras suas relações.

5.3.3 Funcionalidades

Com relação à geração dos gráficos de comparação dos genomas, Syntainia oferece as seguintes funcionalidades através do assistente (Figura 5.4):

- Dois modos de seleção dos arquivos de entrada de dados:
  - Arquivos de resultado do BLAST previamente gerados;
  - Arquivos de sequência em formato FASTA para execução do BLAST — opção que resulta na necessidade de seleção do genoma de referência, ou seja, aquele que será mapeado nos demais;
- Seleção do método de organização dos grupos de genes para a melhor visualização: algoritmos otimista ou realista (ver Seção 5.4);
- Especificação do padrão dos nomes do grupos de genes (expressão regular para os nomes de cromossomos, *supercontigs* e *scaffolds* ou qualquer outra forma de agrupamento que o pesquisador julgar conveniente); e
- Seleção do arquivo de dicionário de dados.

Após a geração do gráfico, o pesquisador pode manipulá-lo através da janela principal do Syntainia (Figura 5.3), na qual estão presentes as seguintes funcionalidades:

- Navegação pelo gráfico gerado;
- Navegação pelos dados do gráfico através de uma árvore;
- Seleção dos genes que devem exibir os caminhos de sintenia (seleção por gene ou por grupo);
- Personalização das cores e da largura das linhas dos caminhos de sintenia, de acordo com a seleção prévia do usuário;
- Mudança manual da ordem dos grupos dentro do genoma;
- Mudança manual da ordem dos genomas;
- Extração de porções do gráfico para visualização dos itens selecionados; e
- Exportação do gráfico para os seguintes formatos: PNG, BMP, JPEG e SVG.

5.4 Algoritmos e Complexidade

Com o objetivo de prover uma melhor visualização dos genomas comparados, Syntainia arranja os grupos de genes em cada genoma. Uma melhor visualização é aquela que tem poucos *caminhos de sintenia* que se cruzam no gráfico. Um *caminho de sintenia* é o traço que liga cada gene a seu respectivo ortólogo entre os genomas comparados. Tomando dois genomas, o número de caminhos de sintenia que se cruzam é minimizado se dois grupos de genes (cada um pertencendo a um genoma) são compostos pelos mesmos genes (ou um subconjunto deles) apresentando posicionamentos relativos bem conservados.

Esse problema é muito semelhante à minimização do número de arestas que se cruzam num grafo desenhado no plano. Muitos estudos tem sido realizados sobre esse problema, sobretudo por se tratar de uma questão recorrente no projeto de circuitos integrados. Em geral, o problema de minimização do cruzamento de arestas num grafo desenhado no plano é NP-difícil, o que tem motivado o desenvolvimento de algoritmos de aproximação para resolver o caso geral, ou que ao menos resolvam casos específicos (Buchheim et al., 2008, 2006; Grigoriev and Bodlaender, 2007).

Voltando ao problema de se obter a melhor visualização, tomando cada gene como um vértice de um grafo (cada gene está ligado aos seus vizinhos imediatos no genoma e ao seu ortólogo no outro genoma, como apresentado na Figura 5.5), o resultado é que se obtém muitas arestas que nunca irão cruzar com outras — as arestas que ligam um gene a outro dentro do mesmo genoma. Além disso, há um grande número de vértices que não podem ser movimentadas no plano, uma vez que os genes que pertençam a um grupo precisam, necessariamente, preservar a ordem dentro do grupo — ordem essa definida pelas posições de início do alinhamento do BLAST e que representam as relações de vizinhança entre os genes.

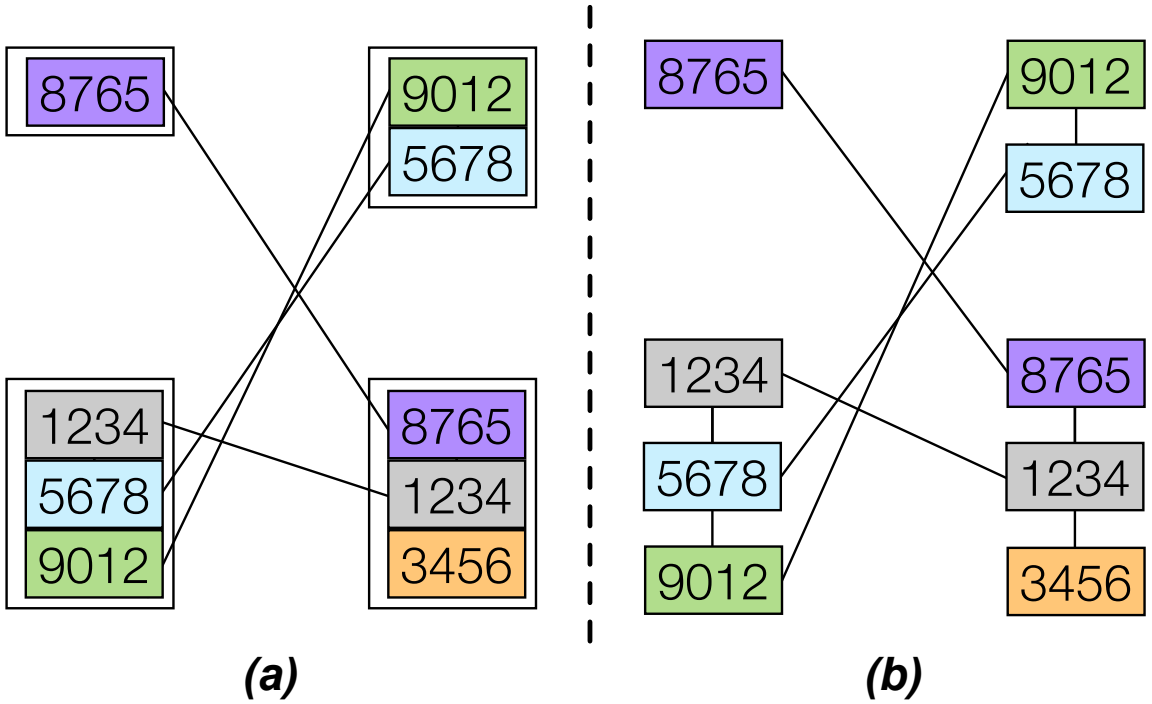


Figura 5.5: Visualização de genes no Syntainia como um grafo. Em (a) é apresentado um exemplo da visualização dos genes pelo Syntainia. Em (b) a visualização é extrapolada para evidenciar as relações entre os genes como um grafo. É importante destacar que ao tentar reduzir o cruzamento de arestas, não é possível fazer um novo arranjo entre os vértices do grupo 1234–5678–9012, por exemplo, uma vez que a ordem dos genes em questão precisa ser preservada. O que pode ser feito é movimentar o grupo 9012–5678 para baixo do grupo 8765–1234–3456 no genoma à direita.

Por último, é altamente desejável que grupos similares estejam dispostos lado a lado, preservando a ordem dos genes quando esses ocorrem em grupos separados entre os genomas.

Esses fatos motivam o desenvolvimento de algoritmos específicos para este caso bem particular do problema de redução do cruzamento de arestas, na expectativa de se encontrar soluções eficazes e eficientes. Assim, dois algoritmos foram projetados para arranjar automaticamente os dados para fornecer uma melhor visualização.

Os algoritmos são baseados na ideia de otimização da visualização entre dois genomas, mesmo se múltiplos genomas são comparados. Considerando a Figura 4.2, os algoritmos tomam pares de genomas da esquerda para a direita, isto é, primeiro consideram Pb18 e Pb01, então Pb01 e Pb03, e assim por diante. Os algoritmos propostos diferem na forma pela qual buscam a ordem dos grupos pertencentes a cada genoma. Essa diferença é explicada a seguir.

5.4.1 Algoritmo Otimista

O primeiro algoritmo (Algoritmo 1) assume que os genes preservam seus posicionamentos relativos entre os genomas comparados. Essa é uma abordagem otimista, uma vez que não há garantias que os genomas comparados são suficientemente similares (isto é, que eles tem grupos semelhantes). Este é um algoritmo rápido se comparado ao outro, porque sua complexidade de tempo é uma função do número de genes n do primeiro genoma G_1 (do lado esquerdo) e o número de genes m do segundo genoma G_2 (do lado direito). Devido ao número de grupos do segundo genoma decrescer a cada grupo similar encontrado, a complexidade de tempo desse algoritmo é $O(nm^2)$.

Algoritmo 1 Algoritmo Otimista

Require: dois genomas G_1 e G_2, cada um com pelo menos um grupo de genes
1: cria uma lista vazia L_1 de grupos de genes
2: cria uma lista L_2 contendo cada grupo de genes do segundo genoma G_2
3: **for all** grupos de genes de G_1 **do**
4: **for all** genes g do grupo **do**
5: **if** o último grupo de L_1 não contém g ou L_1 está vazio **then**
6: **if** existe um grupo em L_2 que contém g **then**
7: coloca o grupo correspondente ao final de L_1 e o remove de L_2
8: **end if**
9: **end if**
10: **end for**
11: **end for**
12: move os grupos remanescentes de L_2 para o fim de L_1
13: substitui a lista de grupos de G_2 por L_1
14: **return** dois genomas, o primeiro igual ao original G_1 e o segundo com a sua lista de grupos de genes do genoma G_2 reordenada

O algoritmo faz com que grupos similares sejam dispostos lado a lado, preservando a ordem dos genes de G_1 em G_2, desde que os grupos sejam suficientemente parecidos.

5.4.2 Algoritmo Realista

O segundo algoritmo (Algoritmo 2) tem uma abordagem baseada em pesos para arranjar os grupos de genes baseado no genoma anterior (que fica do lado esquerdo). Ele não assume que os genomas comparados tem grupos muitos similares. Este é um algoritmos mais lento, se comparado ao primeiro, porque é necessário percorrer todo o segundo genoma a cada passo. Tomando o número n de genes do primeiro genoma G_1 (do lado esquerdo) e o número m de genes do segundo genoma G_2 (do lado direito), a complexidade de tempo deste algoritmo é $O(m^2(n + \log m))$. Essa complexidade pode ser explicada considerando que para cada gene de G_1 é necessário chamar uma sub-rotina de ordenação (Quicksort, por exemplo) e é preciso percorrer a lista que contém o número de ocorrências de genes ortólogos relativos a cada grupo em ordem decrescente.

Algoritmo 2 Algoritmo Realista

Require: dois genomas, G_1 and G_2, cada um com pelo menos um grupo de genes

1: cria uma lista vazia L_1 de grupos de genes
2: cria uma lista L_2 com cada grupo de genes do segundo genoma G_2
3: cria uma tabela T que relaciona um grupo a um contador de ocorrências
4: **for all** grupos de genes de G_1 **do**
5: reiniciar T
6: **for all** genes g do grupo **do**
7: **if** se há um grupo em L_2 que contém g **then**
8: incrementa o contador do grupo correspondente em T
9: **end if**
10: **end for**
11: ordena T em ordem decrescente dos contadores, seguido pela ordem alfabética dos grupos quando o valor dos contadores é igual
12: considerando o primeiro grupo de T, move o correspondente grupo de L_2 para o fim de L_1
13: **for all** grupo em T com contador maior ou igual à metade do tamanho do grupo **do**
14: move o grupo correspondente em L_2 para o fim de L_1
15: **end for**
16: **end for**
17: move os grupos remanescentes de L_2 para o fim de L_1
18: substitui a lista de grupos de G_2 por L_1
19: **return** dois genomas, o primeiro igual ao original G_1 e o segundo com a lista de grupos de genes do genoma G_2 reordenada

O algoritmo faz com que grupos similares sejam dispostos lado a lado, porém, é incapaz de preservar a ordem dos genes de G_1 em G_2. Isso ocorre porque um contador elevado de genes num grupo de G_2 elimina a relação de ordem que poderia haver entre os genes de G_1 em G_2.

5.5 Arquitetura, Estruturas de Dados e Implementação

A arquitetura do software é baseada em duas camadas: interface com o usuário e manipulação dos dados. O subsistema de interface com o usuário é orientado pelas recomendações do Java Swing (Oracle Corporation, 2010b). O subsistema de manipulação dos dados é composto pelos seguintes pacotes: núcleo (algoritmos e *pipeline*), entidades (estruturas de dados), gráficos (módulos de desenho e de exportação para múltiplos formatos de imagem) e analisadores (processamento dos arquivos de entrada, como os resultados do BLAST).

5.5.1 Manipulação dos Dados

As estruturas de dados do Syntainia são apresentadas na Figura 5.6. Essas estruturas de dados são geradas e organizadas dentro de um processamento do tipo *pipeline*, apresentado pela Figura 5.7.

Estruturas de Dados

As classes utilizadas pelo Syntainia para organizar os dados e desenhar o gráfico são agrupadas por: dados genômicos, dados de anotação e dados para desenho (Figura 5.6). Todos essas classes são agrupadas dentro de uma simples classe que pode ser serializada: `SyntainiaComparisionData`. A serialização de objetos permite que as estruturas de dados sejam facilmente salvas em arquivos ou transmitidas via rede.

Os dados genômicos são organizados sob uma lista de genomas (atributo do tipo `ArrayList<Genome>`), que contém ainda um nome. Cada genoma possui um nome e é composto por uma lista de grupos de genes (`ArrayList<GenesGroup>`). Finalmente, cada grupo de genes contém um nome e uma lista dos melhores alinhamentos de um resultado do BLAST (`ArrayList<BestHit>`). Um melhor alinhamento do BLAST (`BestHit`) é formado pelo nome da sequência que está sendo procurada num genoma (`queryName`), o nome da sequência do genoma que foi alinhada com a sequência que está sendo procurada (`subjectName`) e a posição de início do alinhamento entre as duas sequências (`subjectStartValue`). É o `subjectName` que determina o grupo (cromossomo, *supercontig* ou *scaffold*) ao qual pertence a sequência procurada ou, quando aplicável, o gene procurado. E é a partir do `subjectStartValue` que a ordem dos genes é definida dentro dos grupos.

Os dados de anotação são organizados na classe `DataDictionary`, sob um atributo do tipo `Hashtable` da biblioteca padrão do Java (Oracle Corporation, 2010a). Essa tabela tem como chave valores de `queryName` e como objeto elementos do tipo

DictionaryRegistry. Cada objeto do tipo DictionaryRegistry contém um nome curto para exibição do gene no gráfico (shortName) e uma lista de categorias (CategoriesList) às quais pertence o gene. CategoriesList trata-se de um ArrayList<String>, cujos valores são processados de um arquivo tabulado fornecido pelo usuário.

Os dados para desenho são representados pela classe CategoryColorTable, que contém uma tabela (Hashtable) que associa uma cor — classe Color da biblioteca padrão do Java (Oracle Corporation, 2010a) — a cada uma das categorias em DataDictionary. Além disso, também são armazenadas informações sobre como desenhar cada linha que liga um gene a seu ortólogo através dos genomas comparados em objetos do tipo GeneDrawConfiguration, estruturados sob uma tabela Hashtable<String, GeneDrawConfiguration> (um atributo da classe SyntainiaComparisionData), na qual a cada queryName estão associados informações sobre como desenhar seu caminho de sintenia. Tais informações para o desenho consistem em marcar se o caminho de sintenia deve ser desenhado como uma linha ou não (asLine), a largura da linha que liga um gene a seu ortólogo (lineThickness) e a cor da linha (color).

Processamento dos Dados

Para gerar as estruturas de dados descritas anteriormente e organizá-las, de modo que o gráfico gerado tenha a melhor visualização possível, foi implementado um *pipeline* que consiste de quatro passos (Figura 5.7). As três primeiras etapas desse *pipeline* são controladas pela classe GenomesProcessor (Figura 5.8). A primeira tarefa é executar o BLAST para cada genoma, caso já não existam resultados disponíveis. A execução do BLAST é controlada utilizando as facilidades providas pelas classes Process e Runtime da biblioteca padrão do Java (Oracle Corporation, 2010a).

O próximo passo é o processamento dos arquivos de resultado do BLAST para gerar as estruturas de dados do Syntainia. Por enquanto, somente a saída HTML do BLAST é processada. Então, a classe BlastHTMLParser processa os arquivos de resultado do BLAST, gerando uma lista ArrayList<BestHit> com os melhores alinhamentos encontrados, assim como uma lista com todos os valores de queryName para os quais não foi possível alinhar sequências. Outro arquivo analisado é o que contém a informação sobre as categorias genômicas e um nome curto para cada sequência, conforme definidos pelo pesquisador. Este último arquivo tem estrutura tabular, com três colunas correspondentes aos seguintes atributos citados anteriormente: queryName, shortName e CategoriesList. A classe DataDictionaryParser é responsável por analisá-lo.

A tarefa seguinte é estruturar os dados, de modo que os genes fiquem ordenados de acordo com o subjectStartValue dentro de seus respectivos grupos. A responsabilidade por estruturar os dados é da classe GenomeOrganizer. Em seguida, é necessário organizar os dados para obter uma melhor visualização dos genomas no gráfico. Para tanto, utiliza-se um dos dois algoritmos desenvolvidos, de acordo com a seleção prévia do usuário. O Algoritmo 1 é implementado pela classe OptimistAligner, enquanto o Algoritmo 2 está codificado na classe

`PessimistAligner`. A Figura 5.8 apresenta um diagrama UML das classes apresentadas.

Finalmente, o último passo é a geração do gráfico. Foram utilizadas os recursos da API Java 2D para desenhar o gráfico (Oracle Corporation, 2010a). Por meio de uma única classe, `GenomesDrawer` (Figura 5.9), que desenha o gráfico a partir de um objeto `SyntainiaComparisionData` e uma referência para onde desenhar, ou seja, um objeto do tipo `Graphics2D`, da API Java 2D. O mesmo método que faz o desenho para a janela principal do Syntainia também gera o gráfico que será exportado para diversos formatos de imagem (PNG, BMP, JPEG) e de gráficos vetoriais — formato SVG, utilizando o Batik SVG Toolkit (ASF, 2010).

5.5.2 Interface com o Usuário

O subsistema de interface com o usuário possui um conjunto de classes responsável pelos elementos visuais que permitem a interação com o gráfico. É possível interagir com o gráfico através de um painel que contém uma árvore hierárquica `JTree`, no qual os genes (folhas) estão organizados em grupos e estes em genomas. Utilizando recursos de arrastar e soltar ("drag and drop") do Java SE 6 oferecidas através da biblioteca Swing (Oracle Corporation, 2010b) é possível reordenar grupos e genomas através da árvore.

Para viabilizar os recursos de interatividade foi necessário derivar classes de `JTree` e de `TreeModel` (Oracle Corporation, 2010a), com vistas à rolagem automática do painel e semelhança visual com o gráfico desenhado, respectivamente nas classes `GenomesTree` e `GenomesTreeModel`. A classe `GenomesTreeModel` manipula a estrutura de dados apresentada na Figura 5.6 de acordo com os eventos tratados por `GenomesTree`.

O painel que contém a árvore de exibição dos genomas constitui parte da janela principal do Syntainia, da qual também faz parte o painel no qual o gráfico é exibido para o usuário. Assim, a classe `GraphicInteractionPanel` contém um painel privado, `GraphicPanel`, sobre o qual o gráfico é desenhado. Isso é possível pela sobrescrita do método `protected void paintComponent(Graphics graphics)`, definido em `JPanel`, cuja responsabilidade é desenhar a área útil do painel (Oracle Corporation, 2010a).

Isso permite que o gráfico seja desenhado de forma bem eficiente e que qualquer modificação na estrutura de dados seja facilmente replicada para o gráfico exibido. Além disso, desenhar o gráfico sobre um `JPanel` simplifica a captura de eventos, tais como cliques e arrastar e soltar, o que facilita a implementação de mecanismos de interação diretamente sobre a figura.

Também fazem parte do subsistema de interface com o usuário as classes responsáveis pelos painéis do assistente para geração de um novo gráfico. Essencialmente, a classe `NewGraphicWizardController` faz o controle da transição entre os painéis da janela do assistente. A classe `NewGraphicWizardFrame` é responsável pela composição da janela do assistente, na qual os painéis de cada etapa são exibidos de acordo com o *layout* de componentes `CardLayout` do Swing (Oracle Corporation, 2010a,b).

Finalmente, outro conjunto de classes que constitui o subsistema de interface com o usuário é o responsável pela integração com o sistema operacional. A classe `OSVariantsManager` tem como atribuições carregar o *look and feel* do Swing correspondente ao sistema operacional no qual o Syntainia está sendo executado, além de instanciar uma das classes que tratam os demais aspectos de integração com o sistema operacional.

As classes responsáveis por aprimorar os elementos da interface gráfica, de modo a aproximar a usabilidade do Syntainia da experiência nativa de cada sistema operacional são: `GTKVariants`, `MacOSXVariants` e `WindowsVariants`, respectivamente para sistemas UNIX que usam o GNOME, Mac OS X e Windows (Figura 5.10). Essas classes ajustam a estrutura de menus e o desenho de botões às recomendações do projeto de interface gráfica de cada sistema operacional. Nesse sentido, foram seguidas as recomendações de projeto de interface com o usuário do Windows (Microsoft Corporation, 2010), do Mac OS X (Apple Inc., 2010a,b) e do GNOME (Benson et al., 2010).

Em especial, a classe `MacOSXVariants` ainda trata da integração com o Dock[3] e o menu do aplicativo[4], utilizando os recursos providos pela classe `Application`, da biblioteca de extensões da Apple (Apple Inc., 2010b).

[3]Barra que controla o acesso a aplicativos abertos no Mac OS X e que abriga atalhos para aplicativos e documentos.

[4]Elemento da barra de menus do Mac OS X que abriga itens de menu comuns em aplicativos, tais como “Sobre”, “Preferências” e “Sair”.

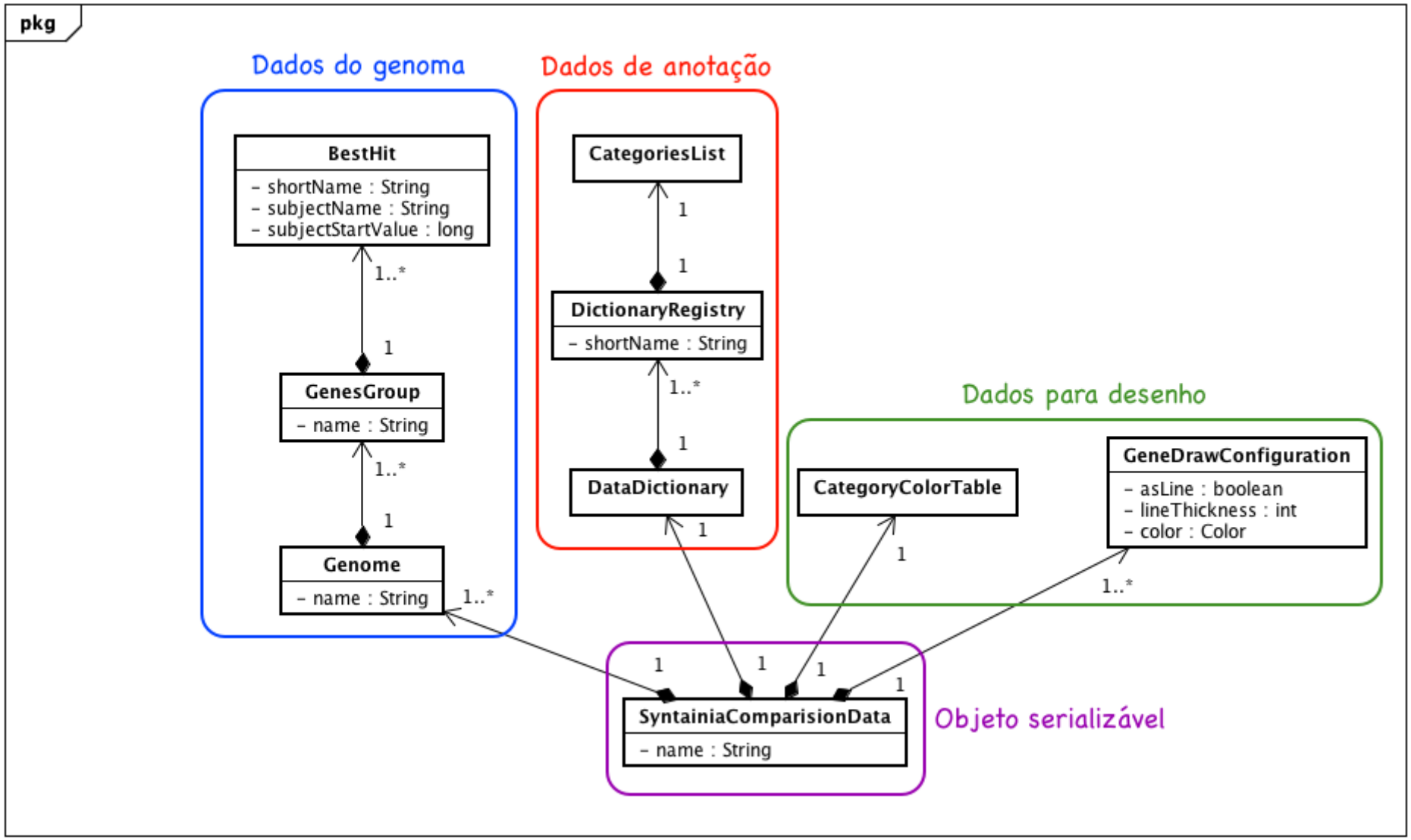


Figura 5.6: Estruturas de dados do Syntainia.

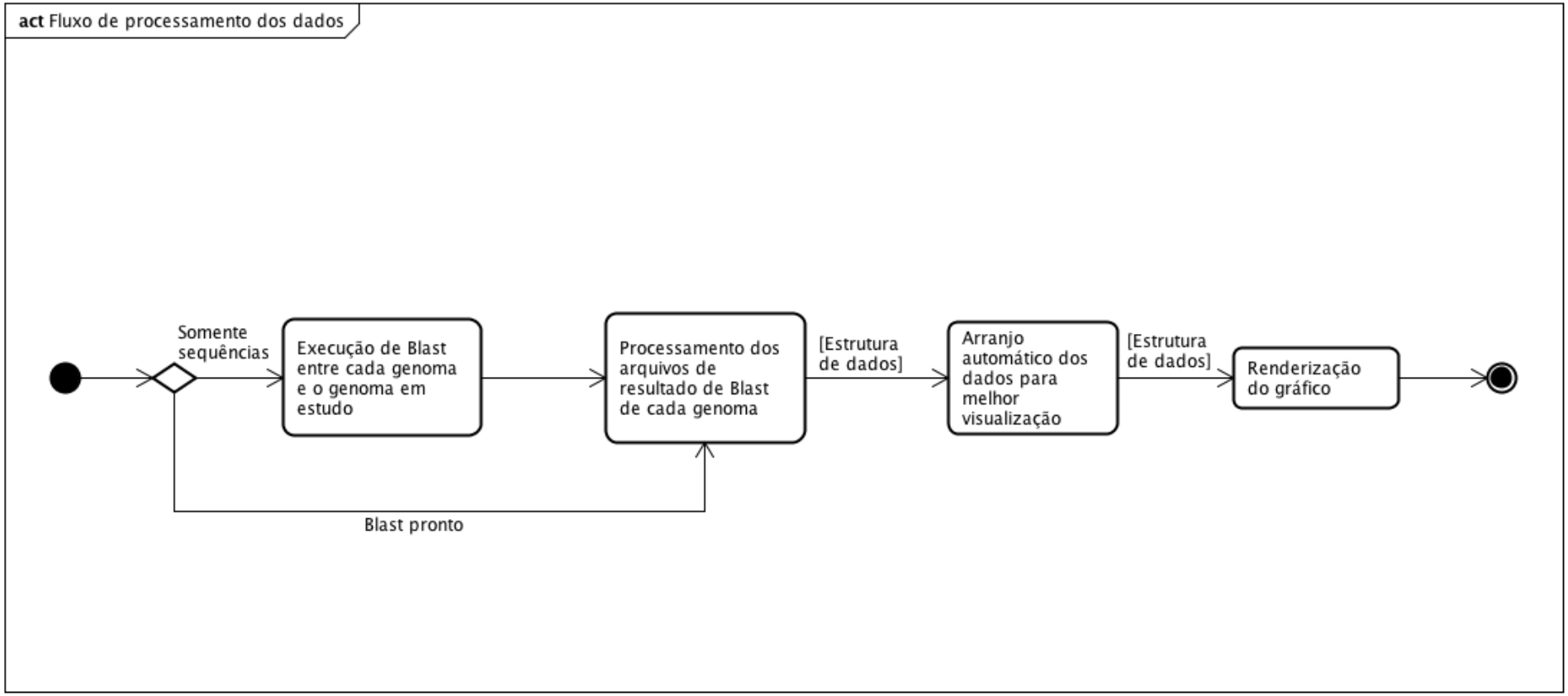


Figura 5.7: Fluxo de processamento dos dados no Syntainia.

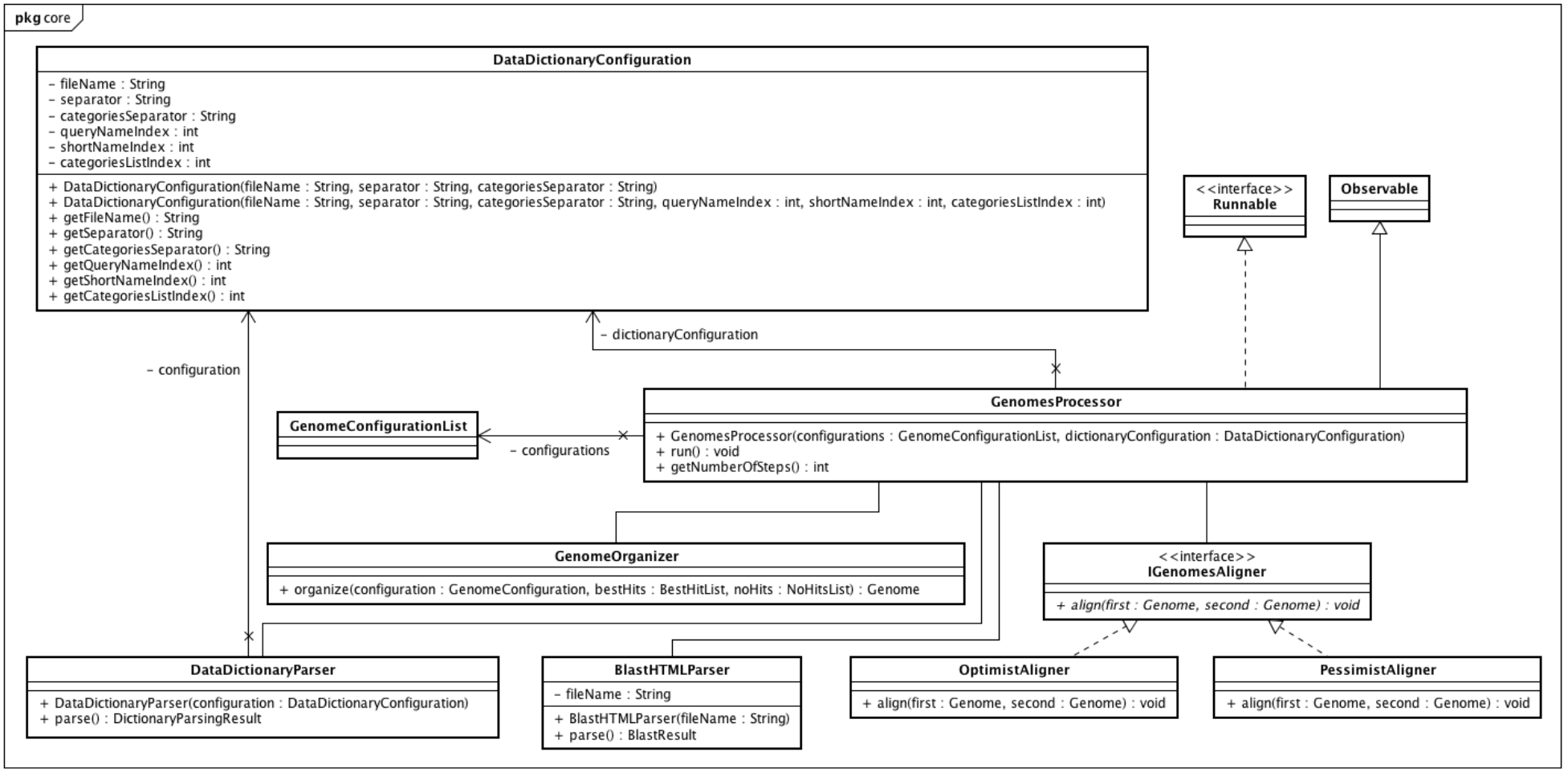


Figura 5.8: Classes do núcleo do Syntainia, responsáveis pela organização das estruturas de dados para geração do gráfico.

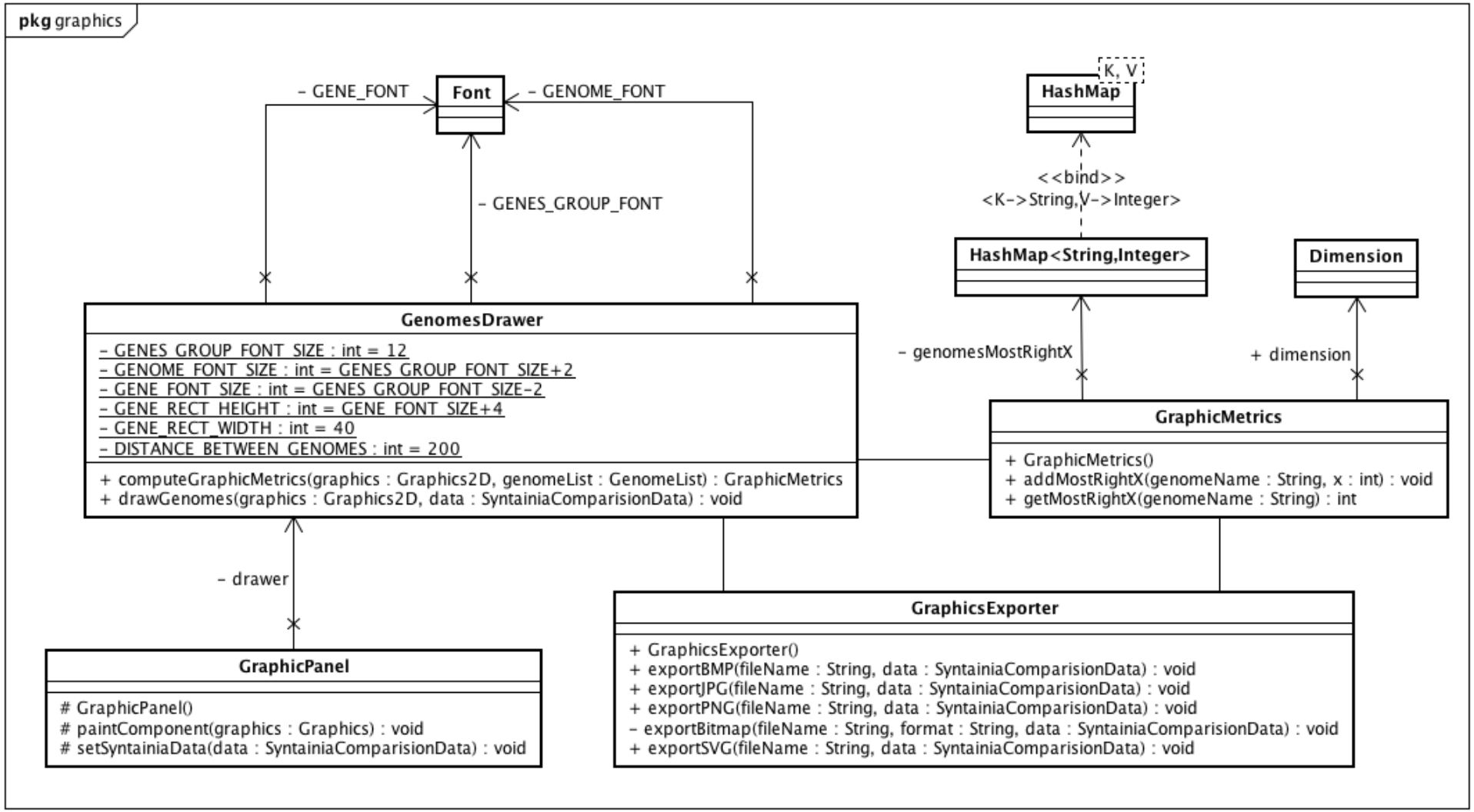


Figura 5.9: Classes para desenho do gráfico, incluindo o painel de exibição na janela principal do Syntainia (`GraphicPanel`) e a classe que exporta o gráfico para diversos formatos (`GraphicsExporter`).

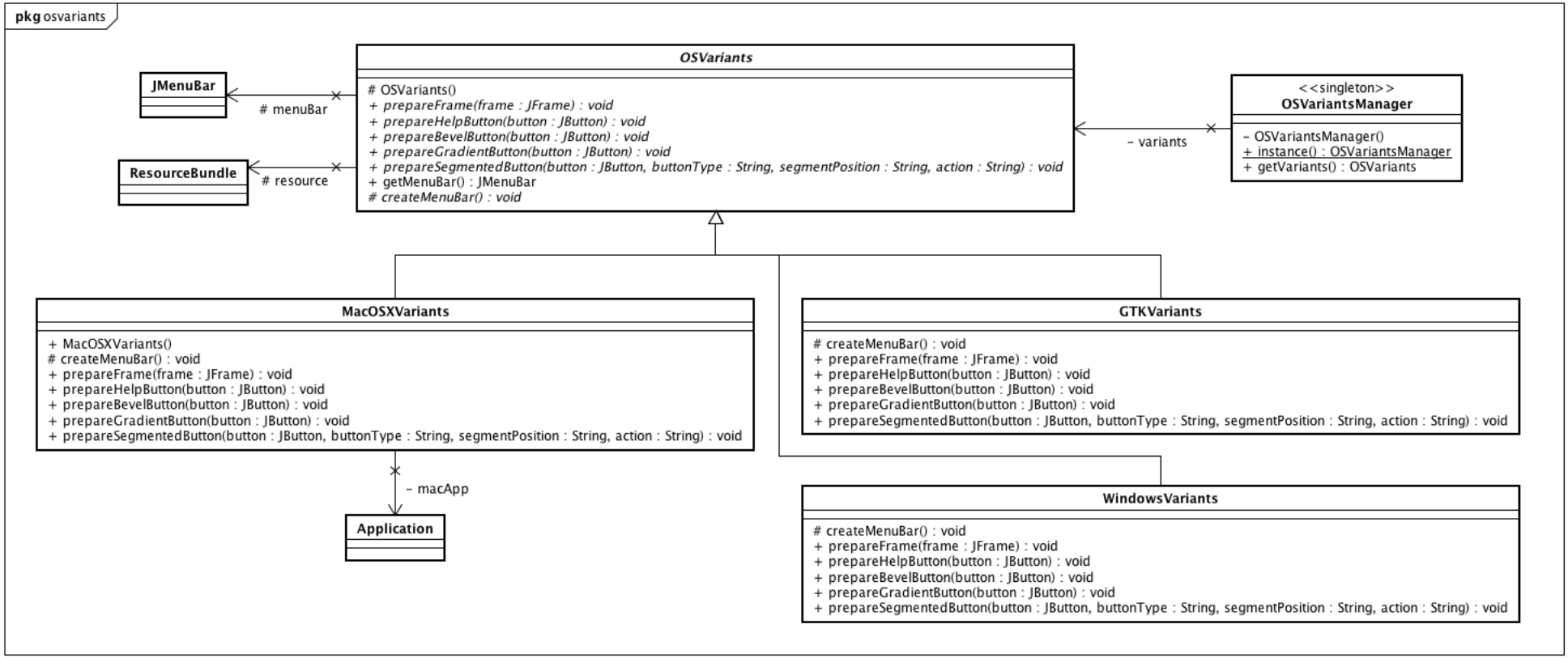


Figura 5.10: Classes responsáveis pela integração com o sistema operacional.

Capítulo 6

Estudo de Caso e Discussão

Este capítulo, na Seção 6.1, apresenta um estudo de caso da utilização do software no trabalho de genômica comparativa do fungo *P. brasiliensis*. Uma breve análise da escalabidade e desempenho da ferramenta é apresentada na Seção 6.2. Finalmente, a Seção 6.3 compara o Syntainia a outros softwares de visualização de sintenias.

6.1 Estudo de Caso

Além de motivar o desenvolvimento do Syntainia, o trabalho realizado por de Carvalho (2010) tem utilizado intensamente essa nova ferramenta, com vistas à obtenção de respostas às perguntas levantadas sobre a conservação de genes de *P. brasiliensis*, isolado Pb01, entre outros dois isolados (Pb03 e Pb18) sequenciados recentemente (Broad Institute, 2010a) e outros fungos (ver Seção 4.2).

Ainda em andamento, o trabalho desenvolvido por de Carvalho (2010) tem como objetivo geral investigar a localização dos genes que podem estar envolvidos nos processos de virulência ou patogenicidade, e genes essenciais, que podem ter relação com o processo de infecção de *P. brasiliensis*. Esta investigação está sendo realizada através de análise comparativa entre os genomas estruturais de Pb01, Pb03, Pb18 e *A. fumigatus*, *A. nidulans*, *C. immitis*, *H. capsulatum*, *C. albicans*, *N. crassa* e *S. cerevisiae*.

A seguir é descrito de forma sucinta o trabalho realizado por de Carvalho (2010), o uso da ferramenta Syntainia e os resultados que já foram obtidos.

6.1.1 Escolha dos Genes e/ou Transcritos

Os genes escolhidos para a localização de seus ortólogos nos genomas selecionados pertencem às categorias: antioxidantes, sinalização celular, alvos para drogas, essenciais, proteínas de choque térmico, Interação patógeno-hospedeiro, ROS/RNI, transporte/MDR e virulência (Felipe et al., 2005a,b).

A escolha dos genes foi baseada nas categorias funcionais às quais pertencem, sendo essas categorias relacionadas, diretamente ou indiretamente, à patogenicidade do *P. brasiliensis*, linhagem Pb01. *Antioxidantes*, *proteínas de choque térmico*, *ROS/RNI* e *transporte/MDR* correspondem a genes que codificam proteínas

geralmente empregadas na proteção do patógeno contra as defesas naturais do organismo hospedeiro ou de substâncias empregadas para o tratamento da doença (de Carvalho, 2010).

As outras categorias consideradas para a seleção são as de genes codificadores de proteínas envolvidas com a manutenção das funções vitais do patógeno como os da categoria *essenciais*. Esses genes são ortólogos aos que, em *C. albicans*, foram experimentalmente comprovados como importantes para o crescimento desse organismo. Outros genes cujas proteínas correspondentes também são importantes para a manutenção do organismo foram categorizadas como pertencentes a *vias de transdução de sinal*; de *virulência*, quando constatados como atuantes diretamente na virulência do patógeno; de *interação patógeno-hospedeiro*, quando importantes para a adesão do patógeno ao organismo que estiver infectando; e de *alvos para drogas*, quando ou o gene existe no genoma do patógeno mas não foi encontrado no genoma humano — consequentemente a proteína codificada por esse gene poderia ser alvo para o desenvolvimento de drogas para neutralizá-la sem afetar as proteínas humanas —, ou as proteínas homólogas possuem diferentes estruturas conformacionais que permitem também usar drogas específicas para as do patógeno (de Carvalho, 2010).

6.1.2 Obtenção das Sequências Genômicas

Todas as sequências genômicas obtidas estão incompletas ou com regiões ainda não definidas ou com espaços ainda não preenchidos — consequentemente, com os cromossomos fragmentados. Algumas sequências como as de Pb18, *A. fumigatus*, *A. nidulans*, *N. crassa*, *S. cerevisiae* e *C. albicans* contém a informação sobre a quais cromossomos pertencem. As demais sequências de Pb03, Pb01, *C. immitis* e *H. capsulatum* ainda não fornecem esses dados, porém, indicam em quais *supercontigs* ou *scaffolds* pertencem (de Carvalho, 2010).

6.1.3 Utilização do Syntainia

O Syntainia gerou gráficos que mostram a ordenação dos ortólogos, representados pelos números PbAESTs, em cada cromossomo ou fragmento de cada um dos genomas dos fungos analisados. Além disso, a ferramenta atribuiu cores a cada PbAEST de acordo com as categorias funcionais aos quais cada um pertencia.

A ordem escolhida de apresentação dos genomas foi baseada na proposta de verificação e comparação da localização dos genes e/ou transcritos de Pb01 em genomas de outros fungos patogênicos e não patogênicos (ver Seção 4.2). Por isso, os três primeiros genomas foram os dos três isolados de *P. brasiliensis*, sendo que o Pb18 ficou sendo o primeiro genoma por ter informações que permitiam associar a que cromossomo pertencia cada um dos fragmentos, identificados originalmente como "*supercont*".

O primeiro gráfico gerado pelo Syntainia mostra todos os dez genomas (Figura 6.1). Devido à grande dimensão do gráfico gerado, de acordo com as análises que se mostraram necessárias, gráficos menores foram gerados, nos quais os isolados de

P. brasiliensis estão juntos ou não dos genomas de fungos patogênicos ou de fungos não patogênicos, como ilustrado pela Figura 6.4.

Assim, com o uso do Syntainia, a tarefa de geração de gráficos para a análise de situações específicas ficou bem simples, uma vez que a partir dos arquivos de resultado do BLAST, previamente gerados, rapidamente a ferramenta gera um gráfico somente com os genomas selecionados. As funcionalidades de realce de caminhos e ocultação de grupos de genes tornaram os gráficos simples e claros, de acordo com as necessidades de cada análise realizada. Finalmente, os gráficos gerados pelo Syntainia foram exportados para o formato SVG, permitindo uma edição mais rica da figura através de programas como o Inkscape[1], sem perda de qualidade.

6.1.4 Organização das Categorias Funcionais

A primeira forma de análise dos resultados obtidos com a geração dos gráficos foi baseada no comportamento organizacional dos genes codificadores das proteínas de cada categoria funcional (anti-oxidantes, transdução de sinal, alvos para drogas, essenciais, HSPs, interação patógeno-hospedeiro, ROS/RNI, transporte/MDR e virulência). O destaque das categorias foi feito inicialmente de forma agrupada, com todas as linhas coloridas de acordo com a categoria dos PbAESTs ligados por elas. Em seguida, cada categoria foi destacada das demais. Nestes gráficos então, só aparecem os PbAESTs e as linhas correspondentes a determinada categoria (de Carvalho, 2010). A Figura 6.2 ilustra essa organização das categorias funcionais, aplicada aos três isolados de *P. brasiliensis*.

6.1.5 Identificação de Sintenias

Os ortólogos aos PbAESTs pertencentes à categoria das HSPs foram todos identificados nos cinco cromossomos de Pb18 e ao longo dos genomas de Pb03 e Pb01. Na Figura 6.1, é possível visualizar ocorrências de sintenias conservadas entre os três isolados de *P. brasiliensis*, bem como casos indicativos de rearranjo cromossomal. Por exemplo, nos fragmentos *chromosome_4.2* de Pb18 e *supercontig_1.2* de Pb03 uma HSP70 (3534/99/3148) aparenta ter sido translocada e invertida, visto que sua posição relativa a outras proteínas de choque térmico nos mesmos fragmentos cromossomais é diferente, bem como na ordem que aparecem os PbAESTs (de Carvalho, 2010).

A categoria das HSPs refere-se a proteínas de choque térmico ("heat shock proteins" — HSPs), chaperonas e às que atuam como co-chaperonas[2]. Essas proteínas de "choque térmico" são algumas das mais conservadas conhecidas e são encontradas tanto em procariotos quanto em eucariotos. Porém, os estímulos para a produção destas e as tarefas que elas realizam variam de organismo para organismo,

[1]Editor de gráficos vetoriais de código aberto, com recursos semelhantes ao Adobe Illustrator ou CorelDRAW e que usa o padrão W3C Scalable Vector Graphics (SVG) como formato de arquivo. O software está disponível para download gratuitamente em `http://www.inkscape.org`.

[2]Chaperonas são proteínas auxiliares, com funções diversas nas atividades celulares, sobretudo na síntese proteica.

e também, com a localização na célula e com a família de HSPs a qual pertencem (de Carvalho, 2010).

A seguir são apresentadas duas análises de sintenias identificadas na comparação entre os 10 genomas estudados. Contudo, o trabalho realizado por de Carvalho (2010) já identificou cerca de 10 conjuntos de genes sintênicos, entre genes de mesma categoria funcional. Como dito antes, a identificação de genes sintênicos entre categorias funcionais distintas ainda é um trabalho em andamento.

ClpX/Mcx1p e HSP60 em Pb18, Pb03, Pb01, *C. immitis* e *H. capsulatum*

O gene codificador para a proteína Mcx1p está representado no transcriptoma do Pb01 por três PbAESTs (179, 586 e 2927) e foi encontrado em nove dos dez genomas analisados, estando ausente em *C. albicans*. Já o gene para HSP60, também representado por três PbAESTs no transcriptoma de Pb01 (269, 318 e 415), está presente nos dez genomas de fungos escolhidos para a comparação neste trabalho. Mas esses dois genes são sintênicos apenas nos três isolados de *P. brasiliensis*, em *C. immitis* e *H. capsulatum* (Figura 6.3). Isso torna esses genes um dos casos de sintenia até então exclusivo de fungos patogênicos da ordem Onygenales (de Carvalho, 2010).

HSP88 e Ssb1 em Pb18, Pb03, Pb01, *H. capsulatum* e *C. albicans*

Os genes codificadores das proteínas Ssb e Sse são representados no transcriptoma do Pb01 por mais de um PbAEST. O Ssb, pelos PbAESTs 783 e 3098 e o Sse pelos PbAESTs 849, 4647 e 4884. Embora essas sequências dos PbAESTs, tanto as do Ssb quanto as do Sse, não tenham sido agrupadas por falta de similaridade entre as sequências nucleotídicas, quando traduzidas, correspondiam à mesma proteína. Com relação aos alinhamentos com os fragmentos genômicos de cada fungo, cada conjunto de PbAESTs alinhavam-se na mesma região, reforçando o fato de se tratarem de um gene só (de Carvalho, 2010).

O gráfico (Figura 6.4) mostra que esses dois genes são sintênicos nos três *P. brasiliensis*, em *H. capsulatum* e *C. albicans*. Mas em *C. albicans* não foi encontrado o PbAEST 4884, um dos PbAESTs correspondentes ao Sse, e os outros dois PbAESTs, 849 e 4647, foram encontrados em cromossomos diferentes. O mesmo ocorre para os PbAESTs do Ssb. Tais casos podem ser exemplos de duplicação e remoção. Duplicação dos genes Ssb e Sse e deleção talvez de um terceiro Sse que seria mais similar ao PbAEST 4884. Assim, há duas cópias de Sse e Ssb sintênicas nos cromossomos 1 e R de *C. albicans* que, nos outros fungos mostrados nesse gráfico, correspondem a uma cópia de cada. O fato de cada uma das cópias estarem no cromossomo R reforça a possibilidade de ter ocorrido uma duplicação e depois uma recombinação intracromossomal desses genes. O cromossomo R de *C. albicans* recebeu essa denominação pela grande quantidade de genes codificadores de RNA ribossomal encontrados neste cromossomo. Em fungos, repetições em tandem contendo rRNA em alguns casos foi considerada como fonte de CLP (*Chromosome Length Polymorphism*), ou seja, facilitam rearranjos gênicos (de Carvalho, 2010).

O caso de inversão só foi observado em Pb01 com os PbAESTs 849 e 4647 (de Carvalho, 2010).

6.1.6 Resultados Parciais

Os resultados até então obtidos mostram genes sintênicos envolvidos com resposta a estresse e enovelamento ou desenovelamento de proteínas (de Carvalho, 2010). A análise sobre a possibilidade de tais genes sintênicos poderem ser explorados por medicamentos que combatam a paracoccidioidomicose é um trabalho ainda em andamento.

Figura 6.1: Todos os 10 genomas comparados pelo Syntainia (de Carvalho, 2010).

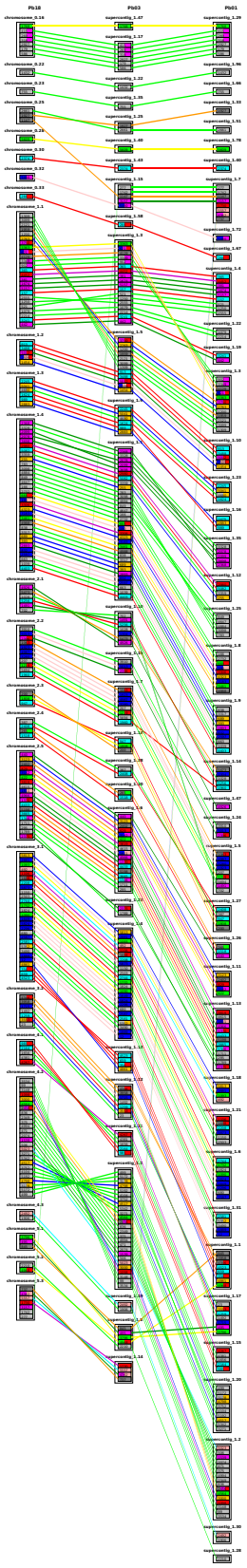


Figura 6.2: Destaque das categorias genômicas entre os três isolados de *P. brasiliensis*. Os caminhos de sintenia estão coloridos de acordo com as categorias genômicas (de Carvalho, 2010).

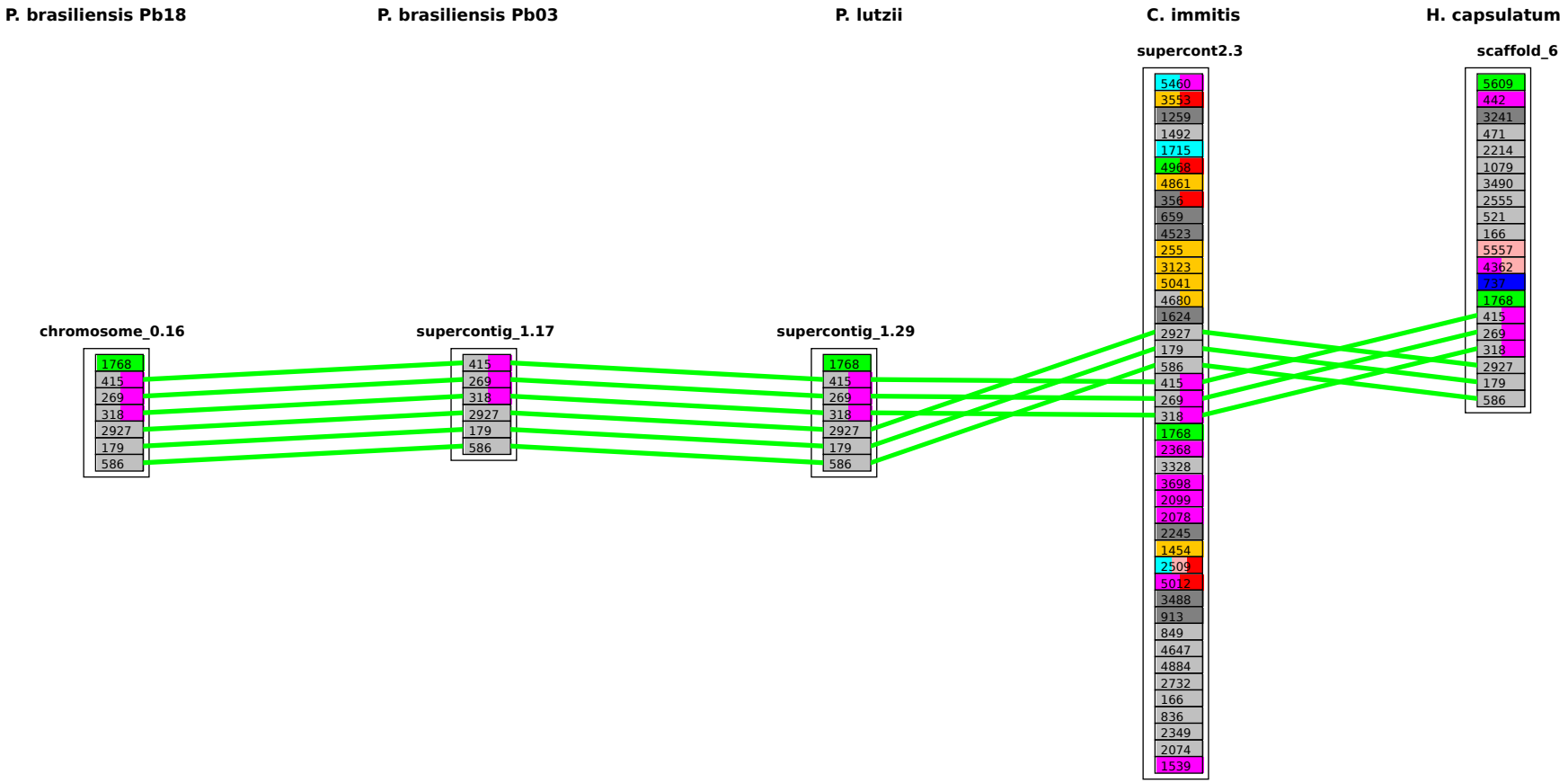


Figura 6.3: Realce de genes sintênicos (*chromosome_0.16*) entre as três linhagens de *P. brasiliensis*, *C. immitis* e *H. capsulatum* (de Carvalho, 2010). No gráfico, o isolado Pb01 já é apresentado com o novo nome proposto: *P. lutzii* (Teixeira et al., 2009).

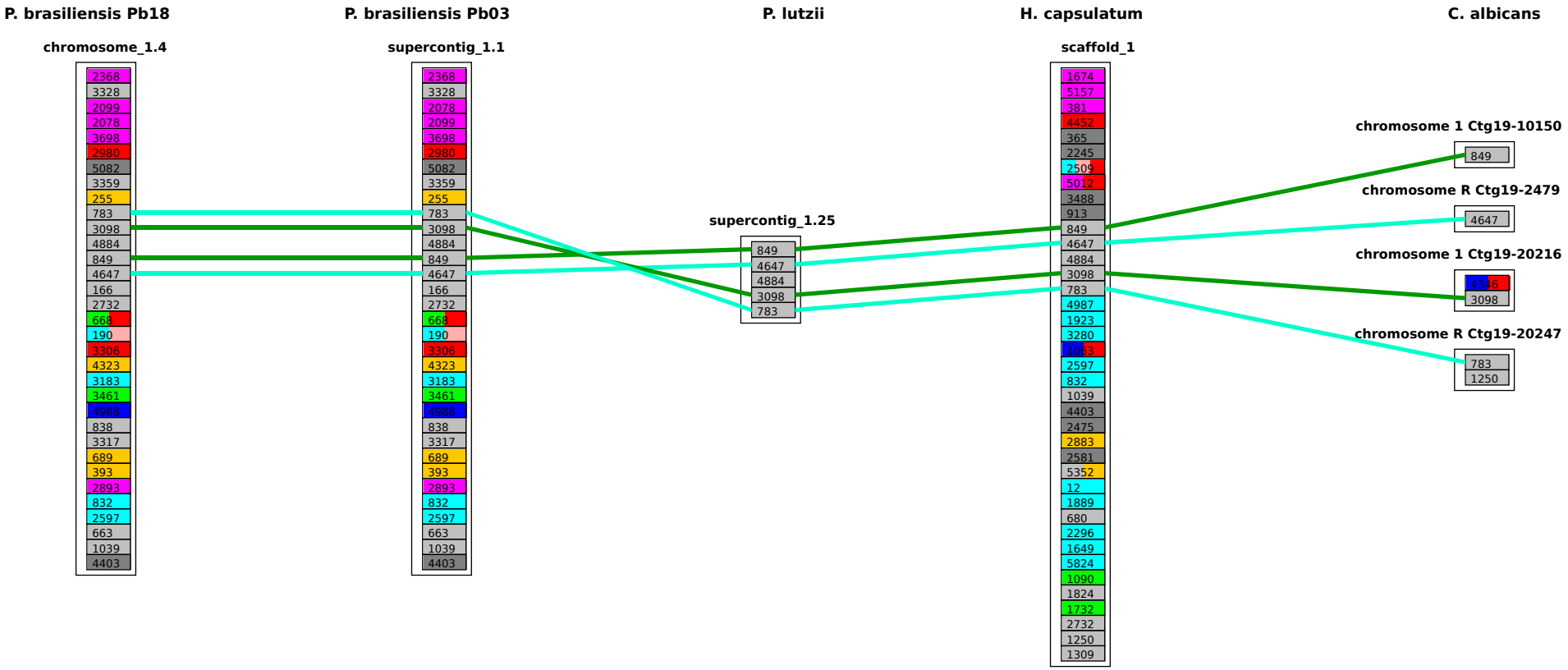


Figura 6.4: Realce de genes sintênicos (*chromosome_1.4*) entre as três linhagens de *P. brasiliensis*, *H. capsulatum* e *C. albicans* (de Carvalho, 2010). No gráfico, o isolado Pb01 já é apresentado com o novo nome proposto: *P. lutzii* (Teixeira et al., 2009).

6.2 Análise de Escalabilidade e Desempenho

Uma análise de escalabilidade é muito importante para softwares de visualização de dados biológicos. Segundo Duboc et al. (2006), escalabilidade é uma propriedade dos sistemas de software caracterizada pelo impacto decorrente de escalar aspectos do ambiente e do projeto de um sistema. Esse impacto pode ser medido sobre outras qualidades, de acordo com aspectos que variam sobre os limites operacionais esperados. Em outras palavras, ao analisar a escalabilidade de um software de visualização é necessário avaliar o volume de dados suportado pela ferramenta, de modo a não prejudicar sua usabilidade.

Por se tratar de um software de visualização de sintenias entre múltiplos genomas, uma análise de escalabilidade do Syntainia essencialmente visa responder à seguinte questão: "*quantos genomas, e de que tamanhos, podem ser analisados pela ferramenta?*". Para tentar responder a essa questão, pode-se aplicar o método proposto por Duboc et al. (2006) ao definir o seguinte cenário:

- **Variáveis escaláveis**: número de genomas, número de grupos (cromossomos, *supercontigs* ou *scaffolds*) em cada genoma e número de sequências do genoma em estudo.
- **Variáveis não-escaláveis**: tamanho da memória da JVM.
- **Variáveis de ruído**: nenhuma identificada.
- **Variáveis dependentes**: consumo de memória.

Nesse cenário, o foco se deu sobre o consumo de memória, uma vez que não foi observada degradação do desempenho durante o processamento dos arquivos e criação das estruturas de dados. O Syntainia foi executado num computador com processador Intel Core 2 Duo de 2,4 GHz e 4 GB de memória RAM, executando a JVM 1.6.0_20 sobre o Mac OS X 10.6.4. Nessa configuração não foram observadas diferenças de tempo de execução significativas, em termos da percepção do usuário, do processamento dos arquivos de resultado do BLAST para a comparação de 2, 3, 5 ou 10 genomas, contra 259 sequências do genoma em estudo, cujo intervalo de tempo médio foi entre 1 e 3 segundos.

Considerando o consumo de memória, para a comparação de 2 genomas contra 259 sequências do genoma em estudo, foram alocados 2,5 MB pela JVM, embora somente cerca de 250 KB tenham sido utilizados pelas estruturas de dados que armazenam as informações necessárias para a produção do gráfico. O consumo de memória é baixo porque não são armazenadas as sequências de cada genoma, mas somente nomes e informações sobre o alinhamento. Foi possível observar que a quantidade de memória usada para armazenar um genoma é inversamente proporcional à quantidade de grupos deste. Por outro lado, a quantidade de memória que armazena todos os genomas é diretamente proporcional ao número de genomas e ao número de sequências do genoma em estudo. Dado que a JVM 1.6.0_20 sobre o Mac OS X 10.6.4, na configuração padrão, utiliza até 123 MB de memória, é possível estimar que, para o mesmo genoma em estudo com 259 sequências, poderiam ser comparados até cerca de 400 genomas, com média de 30 grupos.

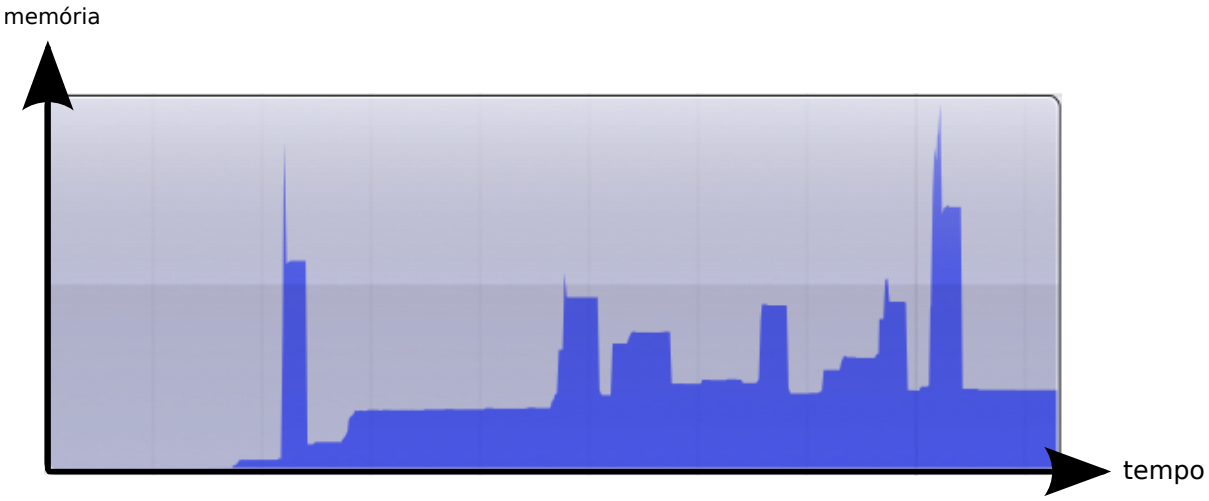


Figura 6.5: Consumo de memória do Syntainia ao longo do tempo. O tempo inicia após a abertura da aplicação. Em seguida é feita a geração de um novo gráfico, comparando dois genomas. O tempo se encerra com a exibição do gráfico. No eixo vertical, o consumo de memória varia de 0 a 14 MB.

Por último, é possível analisar o consumo de memória do Syntainia ao longo do tempo. A Figura 6.5 apresenta um gráfico de consumo de memória do Syntainia, da abertura da aplicação até a exibição de um gráfico novo comparando dois genomas. O primeiro pico de consumo corresponde à abertura do assistente para geração de um novo gráfico. Os picos seguintes correspondem ao processamento dos arquivos e criação das estruturas de dados do Syntainia e, finalmente, ao desenho do gráfico. Após a geração do gráfico, é possível observar que a quantidade de memória usada pela aplicação é baixa (cerca de 2,5 MB). No momento de maior consumo de memória, foram utilizados cerca de 14 MB. Os picos de consumo são explicados pelo fato de a leitura dos arquivos utilizar *buffer* e à necessidade de criação de estruturas auxiliares para a organização da estrutura de dados que armazena todos as informações do gráfico.

6.3 Comparação com Outras Ferramentas

O método de visualização proposto por de Carvalho (2010) não se assemelha muito às formas mais comuns de visualização de comparação entre genomas para a identificação de sintenias. De fato, a visualização implementada pelo Syntainia é bem original, podendo ser comparada diretamente com poucas ferramentas.

Em geral, a maioria das ferramentas de visualização apresentadas na Seção 3.3 apresenta um grande volume de informações, sobretudo relativas a posicionamento dos genes, seja em relação ao genoma completo, seja em relação ao cromossomo do qual faz parte. Por outro lado, Syntainia está focado em mostrar como os genes estão relacionados: sua vizinhança, agrupamento e conservação entre espécies. A aplicação não tem por objetivo apresentar a estrutura dos genomas comparados.

Nesse sentido, ferramentas que apresentam pouca informação diretamente sobre o gráfico, ou que a exibem somente sob demanda são: SyntenyVista (Hunt et al., 2004), Cinteny (Sinha and Meller, 2007), Sybil (TIGR, 2009) e MizBee (Meyer et al., 2009). Em especial, a ferramenta SyntenyVista introduz um modo de visualização denominado *cartoon scaling*, que ignora a estruturação espacial dos genes, tal como a visualização fornecida pelo Syntainia. Cinteny e MizBee possuem mais de um modo de visualização, dos quais os mais simples diferem visualmente do gráfico gerado pelo Syntainia, ou seja, não deixam de omitir dados estruturais sobre os genes. Por outro lado, a ferramenta Sybil, com o seu gradiente de sintenia, apresenta uma forma de visualização bastante simples e com pouca informação, mas que difere completamente de todas as demais ferramentas.

Do ponto de vista dos mecanismos de interação com o gráfico, em geral a grande maioria das ferramentas apresenta alguma forma de seleção do que será exibido e/ou *zoom*. Embora o Syntainia ainda não seja capaz de prover *zoom*, a ferramenta permite que o usuário selecione os genomas e grupos de genes (cromossomos, *supercontigs* ou *scaffolds*) que deseja visualizar.

Por outro lado, Syntainia oferece uma série de mecanismos de interação que só encontram paralelo na ferramenta SyntenyVista (Hunt et al., 2004). Syntainia oferece a possibilidade de o usuário alterar formas e cores dos caminhos de sintenia, funcionalidade não fornecida pelo SyntenyVista ou outras ferramentas. Também é exclusividade do Syntainia a possibilidade de mudar a ordem de grupos de genes e de genomas no gráfico, de modo a simplificar a visualização de acordo com as necessidades do pesquisador. Já o SyntenyVista permite que o usuário movimente um cromossomo ao longo do seu eixo vertical, o que possibilita o alinhamento visual de cromossomos, facilitando assim a visualização das características de conservação entre os grupos de genes alinhados. Além disso, o SyntenyVista fornece também a possibilidade de o usuário inverter um cromossomo, de modo a anular diferenças relativas ao sentido em que foi feito o sequenciamento de cada genoma. Essas duas funcionalidades do SyntenyVista, embora ainda não implementadas pelo Syntainia, são de grande utilidade para os pesquisadores.

Avaliando as questões que um pesquisador espera esclarecer ao estudar os gráficos de comparações de genomas, no contexto da visualização de sintenias, Syntainia não é capaz de fornecer respostas às questões que envolvam aspectos estruturais dos genomas, cromossomos e genes. Isso ocorre justamente porque Syntainia omite essas informações do usuário. Ainda assim, a ferramenta proposta neste trabalho fornece respostas à maioria das questões levantadas por Meyer et al. (2009), assim como a maioria das ferramentas analisadas.

Em termos da escalabilidade do número de genomas comparados pelas ferramentas mais comuns, nem todas são capazes de analisar mais de dois genomas ou, mais precisamente, um número arbitrário de genomas, tal como Syntainia. Assim, as ferramentas que, como o Syntainia, são capazes de comparar múltiplos genomas são: Mauve (Darling et al., 2004), ACT (Carver et al., 2005), SynView (Wang et al., 2006), GBrowse_syn (McKay, 2007), Cinteny (Sinha and Meller, 2007), MEDEA (Broad Institute, 2009) e Sybil (TIGR, 2009). Com as exceções de Mauve e Cinteny, todas as demais ferramentas, inclusive o Syntainia, utilizam BLAST para efetuar as comparações entre os genomas.

Com relação à interface com o usuário, algumas ferramentas apresentam interfaces web, uma vez que são executadas em servidores na rede, com alguns mecanismos sendo executados pela navegador, com o uso de JavaScript ou Adobe Flash. E há também as ferramentas que, assim como o Syntainia, são aplicativos *desktop*, como: Apollo (Clamp et al., 2003), SyntenyVista (Hunt et al., 2004), Mauve (Darling et al., 2004), ACT (Carver et al., 2005) e MizBee (Meyer et al., 2009). Em geral, os aplicativos *desktop* analisados são escritos em Java e não apresentam qualquer preocupação de integração com o sistema operacional, tampouco procuram apresentar uma interface gráfica simples e intuitiva, aspectos que nortearam o desenvolvimento do Syntainia.

A seguir, a Tabela 6.1 apresenta um resumo da comparação entre o Syntainia e as ferramentas discutidas na Seção 3.3. Mais uma vez os softwares são comparados de acordo com os requisitos definidos por Hunt et al. (2004) e os objetivos e tipos de gráficos propostos por Meyer et al. (2009).

Tabela 6.1: Comparação do Syntainia com outras ferramentas de visualização de sintenias, segundo os requisitos definidos por Hunt et al. (2004) e os objetivos e tipos de gráficos propostos por Meyer et al. (2009).

Ferramentas	**Requisitos**								**Objetivos**														**Gráficos**					
	A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10	11	12	13	14	V1	V2	V3	V4	V5	V6
Apollo	•								•	•	•	•	•	•	•	•	•		•	•	•	•	•					
SyntenyView	•	•																	•	•	•	•						•
SyntenyVista	•	•	•	•	•	•	•			•	•	•	•	•	•	•	•		•	•	•	•	•					•
Mauve		•							•	•	•	•	•	•	•	•	•		•	•	•	•	•					
ACT		•							•	•	•	•	•	•	•	•	•		•	•	•	•	•					
SynBrowse	•	•							•	•	•	•	•	•	•	•	•		•			•	•					
SynView	•	•							•	•	•	•	•	•	•	•	•		•			•	•					
GBrowse_syn	•	•							•	•	•	•	•	•	•	•	•		•			•	•					
Cinteny	•	•	•				•		•	•	•	•	•	•	•	•	•		•	•	•	•	•				•	
MEDEA	•	•					•		•	•	•	•	•	•	•	•	•		•	•	•	•	•					
Sybil										•	•	•	•	•	•	•	•		•	•	•	•	•					
MizBee	•		•				•		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•				
Syntainia	•		•	•	•		•	•	•	•		•	•	•	•	•	•		•		•							•

Capítulo 7

Conclusões e Trabalhos Futuros

A visualização de dados biológicos tem se mostrado cada vez mais necessária para a realização das análises dos dados genômicos produzidos pelos projetos de sequenciamento. Em especial, a identificação de sintenias, tarefa comum em análises de genômica comparativa, demanda ferramentas de visualização que deixem claras as relações entre os genes de múltiplos genomas. O método de visualização proposto por de Carvalho (2010) fornece uma visão clara das relações de genes, entre múltiplos genomas, agrupados em cromossomos, *supercontigs*, *scaffolds* ou outro mais conveniente para o pesquisador.

O método proposto por de Carvalho (2010) foi implementado pela ferramenta Syntainia, proposta neste trabalho. Entre as principais características desenvolvidas para a ferramenta, e que são de grande utilidade para os biólogos, destacam-se a automação do processo de comparação dos genomas e geração do gráfico; desenvolvimento e implementação de algoritmos de arranjo automático dos dados para melhor visualização dos genomas; e interação com o gráfico (realce dos caminhos de sintenia, extração de porções do gráfico e re-arranjo dos dados).

Comparado a outras ferramentas de visualização, Syntainia oferece um conjunto equilibrado de funcionalidades e responde às principais questões levantadas por um pesquisador quando este busca a identificação de sintenias. Aplicado ao estudo de caso para visualização de sintenias entre *P. brasiliensis* e outros fungos relacionados, desenvolvido por de Carvalho (2010), Syntainia mostrou-se um software poderoso e flexível, fornecendo os meios necessários para a identificação de sintenias entre os fungos, auxiliando na identificação, até o momento, de 10 conjuntos de genes sintênicos.

Ainda não foram implementados mecanismos que forneçam a capacidade de obtenção de informações sobre os genes a partir da visualização. Não foram implementados também mecanismos de interação direta sobre o gráfico, suporte a desfazer/refazer e *zoom* na área de visualização. Ainda está em estudo qual tecnologia será utilizada para persistir as comparações e os gráficos gerados pelo Syntainia. Uma das alternativas que estão sendo avaliadas é a utilização de um banco de dados relacional embarcado à ferramenta, como o HyperSQL DataBase (Simpson and Toussi, 2010), o que permitirá a produção de relatórios facilmente, por meio de consultas SQL personalizadas. Outra funcionalidade pendente é a possibilidade de alterar as cores das categorias genômicas informadas no dicionário de dados.

Outras funcionalidades desejáveis para o Syntainia são a possibilidade de o usuário inverter um cromossomo, de modo a tornar mais claras diferenças relativas ao sentido em que foi feito o sequenciamento de cada genoma e, assim, possibilitar que o pesquisador dê o tratamento adequado a cada caso; e permitir que o usuário movimente um cromossomo ao longo do seu eixo vertical, o que possibilita o alinhamento visual de cromossomos, facilitando assim a visualização das características de conservação entre os grupos de genes alinhados. Além disso, um recurso que simplificaria a geração dos gráficos seria a capacidade de o Syntainia automaticamente identificar grupos de genes, sem a necessidade de o usuário informar uma expressão regular que defina os nomes dos grupos.

Outros trabalhos futuros relacionados ao desenvolvimento do Syntainia incluem a elaboração de uma opção para processamento distribuído do BLAST, a fim de prover melhor desempenho quando comparando genomas muito extensos; e adição de outros mecanismos de comparação de genomas, diferentes de BLAST. Finalmente, seria interessante o desenvolvimento de uma versão *web* da ferramenta, com a possível integração a sistemas de anotação genômica e/ou *pipelines* de sequenciamento (Coimbra et al., 2007).

Referências

Access Excellence (2010). RNA Ribonucleic Acid — A More Detailed Description. Disponível em http://www.accessexcellence.org/RC/VL/GG/rna2.php (accessado em maio de 2010). 11

Altschul, S. F., Gish, W., Miller, W., Meyers, E. W., and Lipman, D. J. (1990). Basic Local Alignment Search Tool. *Journal of Molecular Biology*, 215(3):403–410. 20, 29

Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402. 1

Alvarez, P. A. (2009). Pipelines para transcritomas obtidos por sequenciadores de alto desempenho. Trabalho de conclusão de curso. Universidade de Brasília, Departamento de Ciência da Computação. 20, 21, 22, 24

Apple Inc. (2010a). Introduction to Apple Human Interface Guidelines. http://developer.apple.com/mac/library/documentation/UserExperience/Conceptual/AppleHIGuidelines/XHIGIntro/XHIGIntro.html. 71

Apple Inc. (2010b). Java Development Guide for Mac OS X. http://developer.apple.com/mac/library/documentation/Java/Conceptual/Java14Development/00-Intro/JavaDevelopment.html. 71

ASF (2010). Batik SVG Toolkit. http://xmlgraphics.apache.org/batik/. 70

Bachhawat, A. K. (2006). Comparative genomics — A powerful new tool in biology. *Resonance*, 11(8):22–40. 25

Benson, C., Clark, B., and Nickell, S. (2010). GNOME Human Interface Guidelines 2.2. http://library.gnome.org/devel/hig-book/stable/. 71

BIOHASKELL (2010). Flower — analysing 454 flowgram files. Disponível em http://blog.malde.org/index.php/flower/ (acessado em maio de 2010). 24

Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M.-C., Estreicher, A., Gasteiger, E., Martin, M. J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S., and Schneider, M. (2003). The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Research*, 31(1):365–370. 7

Broad Institute (2009). MEDEA: Comparative Genomic Visualization with Adobe Flash. Página do projeto na Internet: `http://www.broad.mit.edu/annotation/medea/` (acessada em maio de 2010). 2, 37, 88

Broad Institute (2010a). *Paracoccidioides brasiliensis* Database. Página do projeto na Internet: `http://www.broadinstitute.org/annotation/genome/paracoccidioides_brasiliensis/MultiHome.html` (acessada em março de 2010). 49, 77

Broad Institute (2010b). *Paracoccidioides brasiliensis* Pb01. Imagem disponível em `http://www.broadinstitute.org/annotation/genome/paracoccidioides_brasiliensis/assets/top-graphic_paracocci2.jpg` e acessada em março de 2010. 47

Brígido, M. M., Walter, M. E. M. T., Oliveira, A. G., Inoue, M. K., Anjos, D. A. S., Sandes, E. F. O., Gondim, J. J., de Carvalho, M. J. A., Almeida Jr., N. F., and Felipe, M. S. S. (2005). Bioinformatics of the *Paracoccidioides brasiliensis* EST Project. *Genetics and Molecular Research*, 4(2):203–215. 49

Buchheim, C., Chimani, M., Ebner, D., Gutwenger, C., J¸nger, M., Klau, G. W., Mutzel, P., and Weiskircher, R. (2008). A branch-and-cut approach to the crossing number problem. *Discrete Optimization*, 5(2):373–388. 64

Buchheim, C., Ebner, D., Jünger, M., Klau, G. W., Mutzel, P., and Weiskircher, R. (2006). *Graph drawing: 13th international symposium, GD 2005, Limerick, Ireland, September 12-14, 2005 : revised papers — Volume 3843 of Lecture notes in computer science*, chapter Exact Crossing Minimization, pages 37–48. Springer Berlin / Heidelberg. 64

Carver, T. J., Rutherford, K. M., Berriman, M., Rajandream, M.-A., Barrell, B. G., and Parkhill, J. (2005). ACT: the Artemis comparison tool. *Bioinformatics*, 21(16):3422–3423. 2, 35, 88, 89

Clamp, M., Andrews, D., Barker, D., Bevan, P., Cameron, G., Chen, Y., Clark, L., Cox, T., Cuff, J., Curwen, V., Down, T., Durbin, R., Eyras, E., Gilbert, J., Hammond, M., Hubbard, T., Kasprzyk, A., Keefe, D., Lehvaslaiho, H., Iyer, V., Melsopp, C., Mongin, E., Pettett, R., Potter, S., Rust, A., Schmidt, E., Searle, S., Slater, G., Smith, J., Spooner, W., Stabenau, A., Stalker, J., Stupka, E., Ureta-Vidal, A., Vastrik, I., and Birney, E. (2003). Ensembl 2002: accommodating comparative genomics. *Nucleic Acids Research*, 31(1):38–42. Section: NEW FEATURES — Comparative genome analysis. 2, 34, 89

Coimbra, R. C. M., Santos, S. S., and Walter, M. E. M. T. (2007). Projeto, Implementação e Aplicação de um *Framework* de Código Aberto para Projetos de

Sequenciamento de Genomas. *REIC – Revista Eletrônica de Iniciação Científica*, 7(3). Disponível em `http://www.sbc.org.br/reic` (acessado em fevereiro de 2008). 17, 49, 55, 92

Darling, A. C., Mau, B., Blattner, F. R., and Perna, N. T. (2004). Mauve: Multiple Alignment of Conserved Genomic Sequence With Rearrangements. *Genome Research*, 14(7):1394–1403. 35, 88, 89

Darnell, J. E., Lodish, H. F., and Baltimore, D. (1986). *Molecular Cell Biology*. Scientific American Books, New York, NY. 8

de Carvalho, A. C. P. L. E., Delbem, A. C. B., Romero, R. A. F., and E. V. Simões, G. P. T. (2004). Computação Bioinspirada. In *Jornada de Atualização em Informática — Congresso da Sociedade Brasileira de Computação*, Salvador. 4, 10, 14

de Carvalho, M. J. A. (2010). *Análise Comparativa de Sintenia Entre Genomas de Fungos Não Patogênicos e Patogênicos Humanos*. PhD thesis, Universidade de Brasília, Brasília. Trabalho em andamento. 2, 46, 49, 50, 52, 54, 55, 63, 77, 78, 79, 80, 81, 82, 83, 84, 85, 87, 91

Delcher, A. L., Harmon, D., Kasif, S., White, O., and Salzberg, S. L. (1999). Improved microbial gene identification with GLIMMER. *Nucleic Acids Research*, 27(23):4636–4641. 19

Duboc, L., Rosenblum, D. S., and Wicks, T. (2006). A framework for modelling and analysis of software systems scalability. In *ICSE '06: Proceedings of the 28th international conference on Software engineering*, pages 949–952, Nova York, NY, EUA. ACM. 86

Ewing, B. and Green, P. (1998). Base-Calling of Automated Sequencer Traces Using Phred. II. Error Probabilities. *Genome Research*, 8(3):186–194. 18

Ewing, B., Hillier, L., Wendl, M. C., and Green, P. (1998). Base-Calling of Automated Sequencer Traces Using Phred. I. Accuracy Assessment. *Genome Research*, 8(3):175–185. 18

Faaborg, A. (2007). The Firefox 3 Visual Refresh: System Integration. Artigo no blog pessoal do autor, disponível em `http://blog.mozilla.com/faaborg/2007/10/10/the-firefox-3-visual-refresh-system-integration/`, acessado em março de 2010. 58, 59

Felipe, M. S. S. (2010). Projeto Genoma Funcional e Diferencial do *Paracoccidioides brasiliensis*. Página do projeto `https://helix.biomol.unb.br/Pb/`, acessada em março de 2010. 47, 48

Felipe, M. S. S., Andrade, R. V., Arraes, F. B. M., Nicola, A. M., Maranhão, A. Q., Torres, F. A. G., Silva-Pereira, I., Pocas-Fonseca, M. J., Campos, E. G., Moraes, L. M. P., Andrade, P. A., Tavares, A. H. F. P., Silva, S. S., Kyaw, C. M., Souza, D. P., Network, P., Pereira, M., Jesuíno, R. S. A., Andrade, E. V., Parente, J. A.,

Oliveira, G. S., Barbosa, M. S., Martins, N. F., Fachin, A. L., Cardoso, R. S., Passos, G. A. S., Almeida, N. F., Walter, M. E. M. T., Soares, C. M. A., de Carvalho, M. J. A., and Brigido, M. M. (2005a). Transcriptional Profiles of the Human Pathogenic Fungus *Paracoccidioides brasiliensis* in Mycelium and Yeast Cells. *J. Biol. Chem.*, 280(26):24706–24714. 2, 46, 49, 50, 77

Felipe, M. S. S., Andrade, R. V., Petrofeza, S. S., Maranhão, A. Q., Torres, F. A. G., Albuquerque, P., Arraes, F. B. M., Arruda, M., Azevedo, M. O., Baptista, A. J., Bataus, L. A. M., Borges, C. L., Campos, E. G., Cruz, M. R., Daher, B. S., Dantas, A., Ferreira, M. A. S. V., Ghil, G. V., Jesuino, R. S. A., Kyaw, C. M., Leitão, L., Martins, C. R., Moraes, L. M. P., Neves, E. O., Nicola, A. M., Alves, E. S., Parente, J. A., Pereira, M., Poças-Fonseca, M. J., Resende, R., Ribeiro, B. M., Saldanha, R. R., Santos, S. C., Silva-Pereira, I., Silva, M. A. S., Silveira, E., Simões, I. C., Soares, R. B. A., Souza, D. P., De-Souza, M. T., Andrade, E. V., Xavier, M. A. S., Veiga, H. P., Venancio, E. J., de Carvalho, M. J. A., Oliveira, A. G., Inoue, M. K., Almeida, N. F., Walter, M. E. M. T., Soares, C. M. A., and Brígido, M. M. (2003). Transcriptome characterization of the dimorphic and pathogenic fungus *Paracoccidioides brasiliensis* by EST analysis. *Yeast*, 20(3):263–271. 15, 48, 49

Felipe, M. S. S., Torres, F. A. G., Maranhão, A. Q., Silva-Pereira, I., Poças-Fonseca, M. J., Campos, E. G., Moraes, L. M. P., Arraes, F. B. M., de Carvalho, M. J. A., Andrade, R. V., Nicola, A. M., Teixeira, M. M., Jesuíno, R. S. A., Pereira, M., Soares, C. M. A., and Brígido, M. M. (2005b). Functional genome of the human pathogenic fungus *Paracoccidioides brasiliensis*. *FEMS Immunology & Medical Microbiology*, 45(3):369–381. 46, 47, 50, 77

FSF (2010). GNU General Public License. Disponível em `http://www.gnu.org/licenses/gpl.html` (acessado em março de 2010). 55

Geeknet, Inc. (2010). What is SourceForge.net? `http://sourceforge.net/about`, acessado em março de 2010. 55

General Electric Company (2010). MegaBACE 1000. Disponível em `http://www.gelifesciences.com/aptrix/upp01077.nsf/Content/Products?OpenDocument&parentid=328332&moduleid=166538` (acessado em maio de 2010). 17

Green, P. (2006). Phred, Phrap, Consed. Available at `http://www.phrap.org/phredphrapconsed.html` (accessed in Februrary 2006). 18, 19

Grigoriev, A. and Bodlaender, H. L. (2007). Algorithms for Graphs Embeddable with Few Crossings per Edge. *Algorithmica*, 49(1):1–11. 64

Huang, X. and Madan, A. (1999). CAP3: A DNA Sequence Assembly Program. *Genome Research*, 9(9):868–877. 18, 19

Hunt, E., Hanlon, N., Leader, D. P., Bryce, H., and Dominiczak, A. F. (2004). The Visual Language of Synteny. *OMICS: A Journal of Integrative Biology*, 8(4):289–305. PMID: 15703477. 2, 31, 32, 35, 37, 45, 56, 88, 89, 90

International Human Genome Sequencing Consortium (2001). Initial sequencing and analysis of the human genome. *Nature*, 409:860–921. 12, 15, 16

Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K. F., Itoh, M., Kawashima, S., Katayama, T., Araki, M., and Hirakawa, M. (2006). From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Research*, 34(suppl_1):D354–357. 28

KEGG (2010). KEGG PATHWAY: Photosynthesis — Reference pathway. Gráfico disponibilizado em `http://www.genome.jp/kegg-bin/show_pathway?map00195`. 30

Kent, W. J. (2002). BLAT — The BLAST-Like Alignment Tool. *Genome Research*, 12(4):656–664. 1, 29

Lathe, W. C., Williams, J. M., Mangan, M. E., and Karolchik, D. (2008). Genomic Data Resources: Challenges and Promises. *Nature Education*, 1(13). 27, 29

Lehninger, A. L., Nelson, D. L., and Cox, M. M. (1995). *Princípios de Bioquímica*. Sarvier, São Paulo. 4, 5, 7, 8, 9, 12

Lemos, M. (2004). Workflow *para Bioinformática*. PhD thesis, PUC-Rio, Rio de Janeiro. Disponível em `http://www.maxwell.lambda.ele.puc-rio.br/Busca_etds.php?strSecao=resultado&nrSeq=5928@1` (acessado em maio de 2010). 17

Letunic, I. and Bork, P. (2007). Interactive Tree Of Life (iTOL): an online tool for phylogenetic tree display and annotation. *Bioinformatics*, 23(1):127–128. 28, 29

Lewis, S. E., Searle, S. M. J., Harris, N., Gibson, M., Iyer, V., Richter, J., Wiel, C., Bayraktaroglu, L., Birney, E., Crosby, M. A., Kaminker, J. S., Matthews, B. B., Prochnik, S. E., Smith, C. D., Tupy, J. L., Rubin, G. M., Misra, S., Mungall, C. J., and Clamp, M. E. (2002). Apollo: a sequence annotation editor. *Genome Biology*, 3(12):82.1–82.14. 2, 34

Li, H., Ruan, J., and Durbin, R. (2008). Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11):1851–1858. 24

Lipman, D. J. and Pearson, W. R. (1985). Rapid and sensitive protein similarity searches. *Science*, 227(4693):1435–1441. 29

Loris (2005). Sanger sequencing read display. `http://en.wikipedia.org/wiki/Image:Sanger_sequencing_read_display.gif`. Disponível em `http://en.wikipedia.org/wiki/DNA_sequencer` (accessado em junho de 2006). 17

Mardis, E. R. (2008). Next-Generation DNA Sequencing Methods. *Annual Review of Genomics and Human Genetics*, 9(1):387–402. 1, 20, 21, 22, 23, 25

Mattick, J. S. (2003). *Noncoding RNAs: Molecular Biology and Molecular Medicine*, chapter Introns and Noncoding RNAs: The Hidden Layer of Eukaryotic Complexity, pages 11–32. Springer. Disponível em http://books.google.com.br/books?id=4HFro6mzdH0C&printsec=frontcover#v=onepage&q&f=false (acessado em maio de 2010). 14

Matute, D. R., McEwen, J. G., Puccia, R., Montes, B. A., San-Blas, G., Bagagli, E., Rauscher, J. T., Restrepo, A., Morais, F., Niño-Vega, G., and Taylor, J. W. (2006). Cryptic Speciation and Recombination in the Fungus *Paracoccidioides brasiliensis* as Revealed by Gene Genealogies. *Molecular Biology and Evolution*, 23(1):65–73. 49

McHardy, A. C. (2008). *Bioinformatics, Volume I: Data, Sequence Analysis, and Evolution, vol. 452. Chapter 8: Finding Genes in Genome Sequence*. Humana Press, a part of Springer Science + Business Media, Totowa, NJ. Book doi: 10.1007/978-1-60327-159-2. 1

McKay, S. (2007). Gbrowse_syn. Página do projeto na Internet: http://gmod.org/wiki/GBrowse_syn (acessado em dezembro de 2009). 2, 36, 88

Meyer, M., Munzner, T., and Pfister, H. (2009). MizBee: A Multiscale Synteny Browser. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):897–904. 2, 32, 33, 34, 37, 45, 57, 63, 88, 89, 90

Microsoft Corporation (2010). Windows User Experience Interaction Guidelines. http://msdn.microsoft.com/en-us/library/aa511258.aspx. 71

Miller, J. R., Delcher, A. L., Koren, S., Venter, E., Walenz, B. P., Brownley, A., Johnson, J., Li, K., Mobarry, C., and Sutton, G. (2008). Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics*, 24(24):2818–2824. 24

Morozova, O. and Marra, M. A. (2008). Applications of next-generation sequencing technologies in functional genomics. *Genomics*, 92(5):255–264. 1

NCBI (2006). FASTA format description. Disponível em http://www.ncbi.nlm.nih.gov/blast/fasta.shtml (acessado em fevereiro de 2006). 18

NCBI (2010). NCBI Map Viewer. Captura de tela da página http://www.ncbi.nlm.nih.gov/mapview/maps.cgi?ORG=hum&MAPS=ideogr,est,loc&LINKS=ON&VERBOSE=ON&CHR=X. 28

Nunes, L. R., de Oliveira, R. C., Leite, D. B., da Silva, V. S., dos Reis Marques, E., da Silva Ferreira, M. E., Ribeiro, D. C. D., Ângelo de Souza Bernardes, L., Goldman, M. H. S., Puccia, R., Travassos, L. R., Batista, W. L., Nóbrega, M. P., Nóbrega, F. G., Yang, D.-Y., de Bragança Pereira, C. A., and Goldman, G. H. (2005). Transcriptome Analysis of *Paracoccidioides brasiliensis* Cells Undergoing Mycelium-to-Yeast Transition. *Eukaryotic Cell*, 4(12):2115–2128. 49

Oracle Corporation (2010a). Java™Platform, Standard Edition 6 API Specification. 68, 69, 70

Oracle Corporation (2010b). Trail: Creating a GUI with JFC/Swing. http://java.sun.com/docs/books/tutorial/uiswing/index.html. 58, 68, 70

Paes, H. C. (2009). Isolamento e caracterização funcional do fator de choque térmico (Hsf) do patógeno termodimórfico *Paracoccidioides brasiliensis*. Master's thesis, Instituto de Biologia — Universidade de Brasília, Brasília. 46, 47

Page, R. D. (1996). Tree View: An application to display phylogenetic trees on personal computers. *Comput. Appl. Biosci.*, 12(4):357–358. 28

Pan, X., Stein, L., and Brendel, V. (2005). SynBrowse: a synteny browser for comparative sequence analysis. *Bioinformatics*, 21(17):3461–3468. 2, 36

Pappas Jr., G. (2003). Rede de pesquisa e desenvolvimento de bioinformática do Centro-Oeste. Disponível em http://www.comciencia.br/reportagens/bioinformatica/bio15.shtml (acessado em junho 2006). 15

Passarge, E., Horsthemke, B., and Farber, R. A. (1999). Incorrect use of the term synteny. *Nature Genetics*, 23:387. 1, 26

Pearson, W. R. and Lipman, D. J. (1988). Improved Tools for Biological Sequence Comparison. *Proceedings of the National Academy of Sciences of the USA*, 85(8):2444–2448. 20, 29

Pop, M. and Salzberg, S. L. (2007). Bioinformatics challenges of new sequencing technology. *Trends in Genetics*, 24(3):142–149. DOI of original article: 10.1016/j.tig.2007.12.007. 1

Raymond, E. S. (2000). *The Cathedral and the Bazaar*. O'Reilly Media, Inc., Sebastopol, California. Disponível em http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ (acessado em março de 2010). 55

Reas, C. and Fry, B. (2006). Processing: programming for the media arts. *AI Soc.*, 20(4):526–538. 37

Roche, F. H.-L. (2009). 454 Life Sciences Announces Comprehensive Solution for De Novo Sequencing and Assembly of Increasingly Complex Genomes. Disponível em http://www.roche.com/media/media_releases/med_dia_2009-04-23.htm (acessado em maio de 2010). 21

Salzberg, S. L., Delcher, A. L., Kasif, S., and White, O. (1998). Microbial gene identification using interpolated Markov models. *Nucleic Acids Research*, 26(2):544–548. 19

Sanger, F., Nicklen, S., and Coulson, A. R. (1977). DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5463–5467. 15

Sayers, E. W., Barrett, T., Benson, D. A., Bolton, E., Bryant, S. H., Canese, K., Chetvernin, V., Church, D. M., DiCuccio, M., Federhen, S., Feolo, M., Geer, L. Y., Helmberg, W., Kapustin, Y., Landsman, D., Lipman, D. J., Lu, Z., Madden, T. L., Madej, T., Maglott, D. R., Marchler-Bauer, A., Miller, V., Mizrachi, I., Ostell, J., Panchenko, A., Pruitt, K. D., Schuler, G. D., Sequeira, E., Sherry, S. T., Shumway, M., Sirotkin, K., Slotta, D., Souvorov, A., Starchenko, G., Tatusova, T. A., Wagner, L., Wang, Y., Wilbur, W. J., Yaschenko, E., and Ye, J. (2010). Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research*, 38(suppl_1):D5–16. 27

Schuster, S. C. (2008). Next-generation sequencing transforms todayś biology. *NATURE METHODS*, 5(1):16–18. DOI of original article: 10.1016/j.tig.2007.12.007. 1

Setubal, J. C. and Meidanis, J. (1997). *Introduction to computational molecular biology*. Brooks/Cole Publishing Company, Pacific Grove, CA. 7, 9, 13

Simpson, B. and Toussi, F. (2010). HyperSQL User Guide. Disponível em `http://hsqldb.org/doc/2.0/guide/index.html`, acessado em junho de 2010. 91

Sinha, A. U. and Meller, J. (2007). Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics*, 8(1):82. 2, 36, 88

Stein, L. D., Mungall, C., Shu, S., Caudy, M., Mangone, M., Day, A., Nickerson, E., Stajich, J. E., Harris, T. W., Arva, A., and Lewis, S. (2002). The Generic Genome Browser: A Building Block for a Model Organism System Database. *Bioinformatics*, 12(10):1599–1610. 36

Ströck, M. (2006). An overview of the structure of DNA. `http://en.wikipedia.org/wiki/Image:DNA_Overview.png`. Disponível em `http://en.wikipedia.org/wiki/DNA` (accessado em junho de 2006). 10

Teixeira, M. M., Theodoro, R. C., de Carvalho, M. J. A., Fernandes, L., Paes, H. C., Hahn, R. C., Mendoza, L., Bagagli, E., San-Blas, G., and Felipe, M. S. S. (2009). Phylogenetic analysis reveals a high level of speciation in the *Paracoccidioides* genus. *Molecular Phylogenetics and Evolution*, 52(2):273–283. 46, 49, 84, 85

TIGR (2009). Sybil: Web-based software for comparative genomics. Página do projeto na Internet: `http://sybil.sourceforge.net` (acessado em dezembro de 2009). 2, 37, 88

Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O., Yandell, M., Evans, C. A., Holt, R. A., Gocayne, J. D., Amanatides, P., Ballew, R. M., Huson, D. H., Wortman, J. R., Zhang, Q., Kodira, C. D., Zheng, X. H., Chen, L., Skupski, M., Subramanian, G., Thomas, P. D., Zhang, J., Miklos, G. L. G., Nelson, C., Broder, S., Clark, A. G., Nadeau, J., McKusick, V. A., Zinder, N., Levine, A. J., Roberts, R. J., Simon, M., Slayman, C., Hunkapiller, M., Bolanos, R., Delcher, A., Dew, I., Fasulo, D., Flanigan, M., Florea, L., Halpern, A.,

Hannenhalli, S., Kravitz, S., Levy, S., Mobarry, C., Reinert, K., Remington, K., Abu-Threideh, J., Beasley, E., Biddick, K., Bonazzi, V., Brandon, R., Cargill, M., Chandramouliswaran, I., Charlab, R., Chaturvedi, K., Deng, Z., Francesco, V. D., Dunn, P., Eilbeck, K., Evangelista, C., Gabrielian, A. E., Gan, W., Ge, W., Gong, F., Gu, Z., Guan, P., Heiman, T. J., Higgins, M. E., Ji, R.-R., Ke, Z., Ketchum, K. A., Lai, Z., Lei, Y., Li, Z., Li, J., Liang, Y., Lin, X., Lu, F., Merkulov, G. V., Milshina, N., Moore, H. M., Naik, A. K., Narayan, V. A., Neelam, B., Nusskern, D., Rusch, D. B., Salzberg, S., Shao, W., Shue, B., Sun, J., Wang, Z. Y., Wang, A., Wang, X., Wang, J., Wei, M.-H., Wides, R., Xiao, C., Yan, C., Yao, A., Ye, J., Zhan, M., Zhang, W., Zhang, H., Zhao, Q., Zheng, L., Zhong, F., Zhong, W., Zhu, S. C., Zhao, S., Gilbert, D., Baumhueter, S., Spier, G., Carter, C., Cravchik, A., Woodage, T., Ali, F., An, H., Awe, A., Baldwin, D., Baden, H., Barnstead, M., Barrow, I., Beeson, K., Busam, D., Carver, A., Center, A., Cheng, M. L., Curry, L., Danaher, S., Davenport, L., Desilets, R., Dietz, S., Dodson, K., Doup, L., Ferriera, S., Garg, N., Gluecksmann, A., Hart, B., Haynes, J., Haynes, C., Heiner, C., Hladun, S., Hostin, D., Houck, J., Howland, T., Ibegwam, C., Johnson, J., Kalush, F., Kline, L., Koduru, S., Love, A., Mann, F., May, D., McCawley, S., McIntosh, T., McMullen, I., Moy, M., Moy, L., Murphy, B., Nelson, K., Pfannkoch, C., Pratts, E., Puri, V., Qureshi, H., Reardon, M., Rodriguez, R., Rogers, Y.-H., Romblad, D., Ruhfel, B., Scott, R., Sitter, C., Smallwood, M., Stewart, E., Strong, R., Suh, E., Thomas, R., Tint, N. N., Tse, S., Vech, C., Wang, G., Wetter, J., Williams, S., Williams, M., Windsor, S., Winn-Deen, E., Wolfe, K., Zaveri, J., Zaveri, K., Abril, J. F., Guigo, R., Campbell, M. J., Sjolander, K. V., Karlak, B., Kejariwal, A., Mi, H., Lazareva, B., Hatton, T., Narechania, A., Diemer, K., Muruganujan, A., Guo, N., Sato, S., Bafna, V., Istrail, S., Lippert, R., Schwartz, R., Walenz, B., Yooseph, S., Allen, D., Basu, A., Baxendale, J., Blick, L., Caminha, M., Carnes-Stine, J., Caulk, P., Chiang, Y.-H., Coyne, M., Dahlke, C., Mays, A. D., Dombroski, M., Donnelly, M., Ely, D., Esparham, S., Fosler, C., Gire, H., Glanowski, S., Glasser, K., Glodek, A., Gorokhov, M., Graham, K., Gropman, B., Harris, M., Heil, J., Henderson, S., Hoover, J., Jennings, D., Jordan, C., Jordan, J., Kasha, J., Kagan, L., Kraft, C., Levitsky, A., Lewis, M., Liu, X., Lopez, J., Ma, D., Majoros, W., McDaniel, J., Murphy, S., Newman, M., Nguyen, T., Nguyen, N., Nodell, M., Pan, S., Peck, J., Peterson, M., Rowe, W., Sanders, R., Scott, J., Simpson, M., Smith, T., Sprague, A., Stockwell, T., Turner, R., Venter, E., Wang, M., Wen, M., Wu, D., Wu, M., Xia, A., Zandieh, A., and Zhu, X. (2001). The Sequence of the Human Genome. *Science*, 291(16):1304–1351. 15, 16

Vickers, T. (2007). Nucleotide. `http://en.wikipedia.org/wiki/Image:Nucleotides_v2.png`. Disponível em `http://en.wikipedia.org/wiki/Nucleotide` (accessado em fevereiro de 2008). 8

Wang, H., Su, Y., Mackey, A. J., Kraemer, E. T., and Kissinger, J. C. (2006). SynView: a GBrowse-compatible approach to visualizing comparative genome data. *Bioinformatics*, 22(18):2308–2309. 2, 36, 88