2 Preliminares

Este capítulo possui uma discussão preliminar do contexto biológico necessário para o entendimento e motivação deste trabalho.

Na discussão do contexto biológico serão apresentados os principais conceitos relacionados às biosseqüências, sua importância e utilização na Biologia Computacional. Desta maneira, serão expostas as propriedades deste tipo de dado, como são persistidos e processados atualmente e os problemas decorrentes dos tipos de persistência utilizados. Nos apêndices anexados à este documento são encontradas informações mais detalhadas dos tópicos apresentados neste capítulo.

2.1 Contexto Biológico

O código genético existente em qualquer ser vivo é usado como informação para a produção de proteínas, as quais são responsáveis pela execução de funções especificas. Esta informação genética é chamada pelos biólogos de biosseqüências, pois é representada através de uma seqüência de letras que descrevem as suas propriedades.

As seqüências de DNA são geradas a partir de um processo conhecido como seqüenciamento de DNA. As técnicas e máquinas atuais para seqüenciamento geram seqüências em fragmentos que posteriormente devem ser montados para determinar a ordem dos nucleotídeos na seqüência completa. Existem vários métodos disponíveis para a montagem de fragmentos, e cada um apresenta vantagens e desvantagens [LBC03].

Uma vez obtidas, as seqüências são analisadas pelos pesquisadores. O objetivo é descobrir todos os sinais relevantes na seqüência do DNA, como a presença de genes, por exemplo, e desenvolver meios de usar esta informação para estudo da biologia, medicina, entre outras áreas.

A análise de seqüências através da comparação de seqüências por similaridade tornou-se uma das operações mais importantes na biologia computacional, cujos resultados dão origem a novos tipos de dados biológicos

como as anotações, ou ainda são entradas para muitas outras operações mais elaboradas, como busca de padrões.

Para atacar o problema de comparação e alinhamento entre seqüências surgiu, na década de 90, a família a de programas BLAST [Ncb05, WU05]. Ao considerar o uso de heurísticas como base para os algoritmos de busca de similaridade nas bases de seqüências, os programas desta família trouxeram uma grande melhora nos tempos de respostas. Apesar de existirem outros programas que incluem heurísticas para realizar comparação de biosseqüências, como os programas da família FASTA [PL88], o BLAST, como será chamado neste trabalho, por ser mais rápido, é o mais utilizado pelos biólogos. Por esta razão, realizar melhorias nesses algoritmos ou em estruturas de dados do banco de dados, as quais facilitem ainda mais a execução destes algoritmos, é fundamental para ajudar a resolver este problema [LC00].

A comparação de seqüências é também a base para outras tarefas dos pesquisadores, como a geração de conhecimento biológico através da interpretação aos dados obtidos experimentalmente. Este conhecimento é representado por novos dados conhecidos como anotações. As anotações resultantes da execução de programas de análise são chamadas de anotações automáticas. Os pesquisadores podem gerar também anotações manuais [LSC03].

2.2 Operações sobre Biosseqüências

Com as seqüências em “mãos”, os biólogos enfrentam o desafio de analisá-las no intuito de obter o que é biologicamente importante. Os resultados de tais análises geram mais informações que devem ser persistidas.

A seguir são mencionadas as principais operações e ferramentas utilizadas na biologia computacional sobre seqüências de DNA ou proteínas. O Apêndice B apresenta com maiores detalhes cada uma dessas operações sobre biosseqüências.

Comparação de seqüências

O alinhamento entre seqüências é um arranjo entre duas ou mais seqüências visando descobrir o grau de similaridade entre estas. As seqüências são dispostas umas sobre as outras de maneira a que as colunas respectivas contenham caracteres idênticos ou similares. Se necessário, os buracos podem ser inseridos. Algumas restrições devem ser respeitadas, como por exemplo, um buraco não pode ser alinhado com outro buraco, ou ainda, buracos não podem ser inseridos no inicio e final das seqüências. Na Figura 1, podemos ver um exemplo de alinhamento encontrado em [LC00]. Os traços (-) representam a inserção de buracos, e espaços correspondem a alinhamentos de letras diferentes.

Figura 1 – Alinhamento entre seqüências

Sendo assim, as operações que podem ser realizadas em um alinhamento são: inserção de buracos, alinhamento de letras idênticas e alinhamento de letras diferentes.

Anotações

As anotações representam o resultado de uma das atividades mais importantes dos projetos de seqüenciamento de genoma. É a interpretação e/ou relação que os pesquisadores da área conseguem estabelecer entre os dados obtidos e fontes de dados já existentes e outros experimentos.

O resultado de um novo seqüenciamento pode ser armazenado em uma fonte de dados pública ou em repositório próprio associado ao experimento. Primeiramente é comum que os biólogos registrem dados gerais associados ao experimento, como máquinas e características dos materiais utilizados, e em seguida, executem alguns programas sobre as seqüências obtidas, como por exemplo, os programas que ajudam a descobrir similaridades com seqüências de outros organismos, citados na seção anterior. Assim são geradas anotações resultantes da análise que os pesquisadores fazem sobre as saídas de tais programas.

Montagem de fragmentos

O seqüenciamento de DNA é um processo que determina a ordem dos nucleotídeos em uma amostra. Independente da técnica, a maior sub-cadeia de DNA com qualidade que pode ser determinada em um procedimento em laboratório possui cerca de 600 a 700 bases [LMM+04].

Desta forma, para seqüenciar um genoma inteiro, o DNA precisa ser dividido em vários fragmentos pequenos que são seqüenciados individualmente. A partir de vários fragmentos de seqüências, busca-se reconstituir o trecho de DNA dos quais esses fragmentos provieram através de comparações entre eles. Este problema é conhecido como montagem de fragmentos (i.e. em inglês, *sequence assembly)* [LBC03]. Existem vários métodos disponíveis, e cada um apresenta vantagens e desvantagens.

2.3 Persistência de Biosseqüências

As biosseqüências ao serem armazenadas recebem uma identificação única e passam a fazer parte de um banco de dados de biosseqüências, público ou privado. A partir daí podem ser usadas como informação para catalogar outras seqüências geradas, ser analisadas com outras ferramentas ou comparadas com seqüências de outros repositórios.

Os principais bancos de biosseqüências existem desde a década passada, e estão em constante crescimento. Por exemplo, o SwissProt [SWI05], um banco curado[1] de proteínas, que em 2001 (Release 40) registrava 101.602 entradas, atualmente contém 194.317 seqüências representando um crescimento de 90% somente nos últimos 4 anos. No *site* do GenBank [Ncb05c] é contabilizado atualmente[2] quase 100 milhões de seqüências com 90 bilhões de bases, sendo que o banco vem dobrando de tamanho a cada 18 meses desde o seu inicio[Ncb05c]. Existem outros bancos públicos como o DDBJ [DDB05] e EMBL [EBI05]. Uma descrição dos bancos de seqüências mais populares pode ser encontrada no Apêndice C.

[1] Os arquivos curados contém dados que foram validados. A manutenção exige muito trabalho manual para comprovar fonte dos dados e até a exigência de verificação experimental em bancada.

[2] Informação obtida no *site* do NCBI em julho de 2005.

A maioria dos bancos de dados está disponível em arquivos texto semi-estruturados. Este formato foi adotado inicialmente pela possibilidade de ser manipulado tanto pelas máquinas quanto pelas pessoas. Porém, dada a diversidade de formatos e códigos, e também pelo volume de dados, isto acaba sendo uma difícil tarefa para os pesquisadores da área hoje em dia.

Alguns dos formatos mais usados são ASN.1 [ASN05], XML [XML05] e FASTA [PL88]. Também existem bancos mantidos em gerenciadores de bancos de dados relacionais como o Oracle [SCD+05], PostgreSQL [Bla05] e MySQL[GOD05] que incluem novos tipos de dados e operadores na tentativa de dar tratamento adequado às biosseqüências. Mais detalhes sobre os formatos de arquivo usados para persistir as biosseqüências podem ser encontrados no Apêndice C e em [FAB05].

Vale ressaltar também que as seqüências podem ser armazenadas com formatos específicos ou intermediários para execução de alguns aplicativos. Um exemplo importante é o pré-processamento necessário para a execução do BLAST, feito através da ferramenta FORMATDB [For05]. O FORMATDB cria arquivos binários a partir de um arquivo de seqüências em formato FASTA ou ASN.1 contra os quais serão comparados a seqüência de consulta durante a execução do BLAST. Três arquivos são gerados: um com seqüências, outro com a informação do cabeçalho identificador de cada seqüência, e o terceiro com uma estrutura de indexação para o acesso aleatório aos arquivos anteriores. Para execução da família FASTA, os dados também são estruturados como no formato FASTA.

2.4 Problemas com o armazenamento de biosseqüências

Ainda não há um padrão para armazenamento de seqüências biológicas, tanto em termos de modelo de dados quanto mecanismo de persistência.

Muitos dos problemas de armazenamento de dados científicos são similares a problemas de banco de dados do ambiente convencional. Contudo, dados científicos possuem algumas particularidades a pouca freqüência de atualização a que são submetidos e serem mantidos por tempo indefinido. Sendo assim, fatores como controle de concorrência e eficiência nas transações, problemas críticos em outras áreas, ainda não são fundamentais para a área

cientifica. Por outro lado, é essencial um processamento de consultas flexível para a maioria das aplicações [SLL00].

Outros problemas são inerentes ao modelo adotado, como no caso do modelo relacional, que não conta com tipos de dados próprios para representar dados específicos, como, por exemplo, dados de biosseqüências.

Entre os formatos específicos criados para a execução de programas de análise de seqüências também existem problemas. Por exemplo, o aplicativo FORMATDB, apresenta limitações para tratar com grandes volumes de dados. Devido a detalhes de implementação, internamente o FORMATDB só pode tratar arquivos com até 4 GB. Este limite equivale a um arquivo com aproximadamente 16 bilhões de caracteres no caso de bases de nucleotídeos ou 4 bilhões para aminoácidos [BKY03]. Para contornar este problema, algumas implementações, como o NCBI-BLAST e a versão comercial do WU-BLAST, contam com um recurso chamado de “alias” de banco de dados ou criação de banco de dados virtuais, que permite combinar múltiplos bancos de seqüências.

Também é importante notar que a forma como as biosseqüências são armazenadas proporciona a ocorrência de redundância de informação. A mesma seqüência pode estar repetida com identificadores diferentes. Não há forma de evitar tal duplicação ou impor restrições, a não ser impondo mais controle na forma de submissão de seqüências aos bancos.

Dados os problemas levantados aqui, questionamentos relacionados à persistência mais adequada para as biosseqüências se fazem necessários, como por exemplo:

1. Porque as biosseqüências continuam sendo persistidas em arquivos textos, se sistemas gerenciadores de banco de dados são mais eficientes para armazenar grandes volumes de dados.

2. Ao optar por sistemas gerenciadores de bancos de dados, deve-se responder quais as estruturas mais adequadas entre as estruturas disponíveis como CLOB ou simplesmente cadeias de texto.

3. Como lidar com o volume de dados de biosseqüências crescente.

Essas questões ajudam a concluir que ferramentas especializadas para a gerência de dados para biosseqüências de maneira eficiente se tornam uma

questão de pesquisa importante dentro do contexto da bioinformática. Os desafios para este tipo de gerenciamento eficiente incluem mecanismos eficientes para armazenamento em memória secundária, métodos de acesso e políticas de gerencia de memória principal.

Particularmente, as técnicas de banco de dados são adequadas na busca de uma solução para este contexto, pois fornecem mecanismos eficientes no gerenciamento de dados. No entanto, é necessária a busca de uma solução especifica para o tratamento de biosseqüências visto que as atuais técnicas de banco de dados não apresentam uma solução definitiva para o problema de persistência e acesso às biosseqüências.

2.5 Trabalhos relacionados

Até onde foi investigado, como também pode ser visto no apêndice C, os trabalhos relacionados com melhoria de eficiência na persistência e acesso a biosseqüências não são extensos. Supõe-se que a característica de multidisciplinariedade advinda da pesquisa em Bioinformática dificulta a compreensão do problema e existência de soluções. Dentro deste contexto foram encontrados alguns trabalhos que lidam com particularidades deste problema.

Em [LL03] foi estudado um mecanismo de gerência de memória para comparação de biosseqüências realizadas pelo programa BLAST. Neste trabalho é apresentada uma estratégia para realizar o acesso às seqüências de uma maneira mais eficiente utilizando uma estrutura circular em memória. Esta estrutura permite iniciar a comparação de uma seqüência de entrada com o banco de seqüências a partir de qualquer posição da seqüência e habilitando seu compartilhamento com outras seqüências de entrada. Desta forma, vários processos BLAST poderiam compartilhar a mesma seqüência do banco de dados fazendo reuso eficiente da memória. Esta estratégia está focada somente no acesso aos dados e não propôs nenhuma mudança na forma de armazenamento das seqüências.

Em [DBL+00] são apresentadas mudanças no algoritmo que melhoraram consideravelmente o desempenho do programa BlastP da família NCBI-BLAST, baseadas em problemas encontrados durante o estudo detalhado do código do programa. Este trabalho foi construído sobre uma versão antiga do Blast, a

versão 1.4, tanto que uma das otimizações que propõe retardar o processo de extensão até que dois casamentos (i.e. em inglês *hits*) na mesma diagonal sejam encontrados, já foi inclusive incorporada à versão atual do Blast.

Outras idéias de melhoras em programas para comparação de biosseqüências apóiam-se na própria natureza *multithread* do sistema como em [Cos02], que apresenta uma estratégia de balanceamento de carga para execução em paralela do BLAST baseado em MPI com os resultados satisfatórios em vários cenários de execução.

Uma estratégia mais geral é estender os sistemas de banco de dados tradicionais para incorporar estruturas e operadores da biologia. Dentre estas estratégias, o SGBD Oracle 10g merece uma atenção particular devido ao grande número de funcionalidades incluídas nesta versão. O Oracle 10g incorpora o BLAST, versão NCBI 2.0, no banco de dados, permitindo a realização das operações disponíveis em todas as versões do BLAST como também consultas em linguagem SQL para pré-filtrar as seqüências ou ainda pós-processar os resultados obtidos [SCD+05]. As operações de busca de similaridade e alinhamentos são feitos através de consultas utilizando uma tabela-função na cláusula FROM de uma sentença SQL. As seqüências devem estar armazenadas no Oracle como um tipo CLOB, ou podem ser mantidos fora do banco de dados.

Outra tática semelhante foi colocada em prática utilizando o gerenciador de dados PostgreSQL [Pos05]. Esta implementação chamada de BlastGres [Bla05] também incorpora o programa BLAST ao gerenciador de banco de dados. Além disto, foram criados novos tipos de dados para representar segmentos de seqüência em conjunto com um novo tipo de índice [HPY05] para acelerar o acesso a uma região de uma seqüência e as características correspondentes. Nesta abordagem as seqüências ainda são guardadas como cadeias de caracteres.

Apesar das estratégias de extensão de SGBDs trazer para dentro do banco de dados as biosseqüências, em ambos os casos as biosseqüências são tratadas como cadeia de caracteres.

Vale lembrar algumas propostas para persistir as biosseqüências como árvores de sufixos apresentadas em [BH04, THP04]. Entretanto, como estes trabalhos exigiram reformulações muito grandes nos programas que

implementam as operações mais freqüentes sobre os dados de biosseqüências, como, por exemplo, a recodificação das funções de acesso aos dados, torna-se difícil o uso prático das mesmas. No inicio deste trabalho, foi feita a tentativa da utilização da estrutura de árvore de sufixo com o BLAST como estratégia de persistência, porém dada a complexidade da aplicação do ponto de vista de implementação será necessário um tempo muito grande.

Deve ser ressaltado o trabalho de Grumbach e Tahi[GT93] relacionado à mudança de formato na persistência de biosseqüências. É apresentado um programa específico para compactação de seqüências de nucleotídeos, utilizando um algoritmo semelhante ao LempelZiv[ZL77] baseado em substituições. O objetivo deste trabalho é explorar as repetições de padrões nas biosseqüências, como a presença de palíndromos. Os resultados obtidos pelo **Biocompress** representaram melhoras consideráveis na taxa de compressão em relação aos algoritmos clássicos, porém o desempenho em geral ainda não é satisfatório.

2.6 Conclusão

Neste capítulo foi apresentada a importância dos dados de biosseqüências para a pesquisa da biologia.

A segunda seção apresentou as principais operações sobre biosseqüências e alguns dos programas associados aos tipos de operações. Foi possível observar que a comparação de seqüências é uma das principais operações para desvendar as funções de biosseqüências ainda desconhecidas. O programa da família BLAST é um dos programas mais utilizados para efetuar esse tipo de tarefa.

Na terceira seção, foram enumeradas as estratégias de persistência das biosseqüências e suas características, podendo obter-se maiores detalhes sobre este tipo de dados no Apêndice A deste trabalho. Foi percebido que embora o volume de dados de biosseqüências seja crescente, a maior parte dos dados de biosseqüências continua sendo armazenada em arquivos textos.

O capítulo foi finalizado apresentando os trabalhos relacionados com persistência e acesso aos dados de biosseqüências. Foram apresentadas três diferentes estratégias de otimização para lidar com o problema da persistência,

as quais foram: gerência de memória, extensão de SGBDs existentes e compactação de biosseqüências.

Em resumo, observou-se que embora existam estratégias para lidar com o problema da ineficiência no processamento de biosseqüências, nenhum dos trabalhos relacionados apresentou uma abordagem integradora. Ou seja, uma abordagem que abrangesse uma estratégia para persistência especifica de biosseqüências em conjunto com um método de acesso adequado a esta forma de persistência e uma gerência de memória que coordenasse este processo.

Desta maneira, no próximo capítulo será investigado um dos programas da família BLAST com o intuito de descobrir o comportamento do mesmo em relação às operações de leitura de dados de biosseqüências.