8 Conclusão

8.1 Contribuição

O crescente volume e a distribuição de dados e processos em Bioinformática torna cada vez mais fácil as descobertas de novas informações biológicas. Entretanto, como são inúmeras as análises que podem ser feitas, os pesquisadores precisam cada vez mais da ajuda de sistemas que os auxiliem em seus trabalhos, especialmente quando estes diversos dados e processos precisam ser combinados formando-se o que chamamos nesta tese de um workflow de Bioinformática.

Neste contexto, esta tese justificou a importância, levantou os requisitos e apresentou uma proposta de um *framework* para um sistema de gerência de análises em biossequências (SGABio), composto por dois sub-sistemas. O primeiro é um sistema de gerência de workflows de Bioinformática (SGWBio), que auxilia os pesquisadores na definição, validação, otimização, execução, monitoramento e agendamento de workflows necessários para se realizar as análises. O segundo é um sistema de gerência de dados em Bioinformática (SGDBio), que trata do armazenamento e da manipulação dos dados envolvidos nestas análises.

O *framework* pode ser instanciado em diversos tipos de ambientes de trabalho dos pesquisadores, desde um ambiente pessoal que possui apenas uma máquina, até ambientes mais complexos que possuem um parque de máquinas, como os ambientes de laboratório e de comunidade. Os principais sistemas existentes estão focados na execução de workflows em um determinado tipo de ambiente. Por exemplo, o myGrid e o Proteus executam seus workflows em *grid*, que é uma das potenciais instanciações do *framework* proposto. De fato, os ambientes de comunidade podem ser implementados sobre uma infra-estrutura de *grid* ou utilizando tecnologia de *Web services*.

O *framework* inclui um gerenciador de ontologias capaz de armazenar ontologias de processos de Bioinformática. De fato, atualmente existem algumas

ontologias utilizadas em Bioinformática, que podem ser classificadas em duas categorias. A primeira provê definições dos processos de Bioinformática, de seus recursos e de seus dados de entrada e saída, como a ontologia proposta nesta tese e as definidas pelos sistemas myGrid e Proteus. A segunda provê definições consistentes de conceitos de Biologia Molecular, como o Gene Ontology e o Ecocyc.

Uma ontologia de processos de Bioinformática define classes que referem-se a processos, recursos e dados comumente envolvidos em análises de biossequências. A diferença principal da ontologia deste trabalho para as definidas pelos sistemas myGrid e Proteus é que ela possui informações mais detalhadas sobre os processos, recursos, dados e processadores, que permitem que o sistema possa:

- auxiliar melhor o pesquisador durante a definição do seu workflow;
- incluir processos auxiliares automaticamente em um workflow;
- alocar os processos, dados e recursos em processadores mais adequados (nos casos dos ambientes de laboratório e comunidade);
- executar o workflow de forma otimizada.

O auxílio para a definição do workflow feito por sistemas como o myGrid e o Proteus estão baseados na identificação de programas de Bioinformática através de suas classificações por tarefas e de seus dados de entrada e saída. Por exemplo, existe a tarefa alinhamento de sequência, que está associada aos processos BLAST e FAST. A ontologia proposta aqui, além de ajudar os pesquisadores desta maneira, possui ainda outras informações que permitem um auxílio mais sofisticado, por exemplo, através de propriedades de qualidades que definem e diferenciam os processos, como desempenho, popularidade, *default*, custo, fidelidade e adequação. Desta forma, o pesquisador pode descobrir qual é o programa em Bioinformática para fazer alinhamento de sequências mais adequado informando que deseja o mais popular, o mais rápido, o mais adequado para projeto de ESTs, etc.. Isto permite que o pesquisador crie um workflow mais adequado para sua análise sem ter um grande conhecimento em Bioinformática.

A inclusão automática de processos tem diversos objetivos. Primeiro, ela pode tornar uma composição de processos coerente através de processos transformadores de formatos. Segundo, ela pode melhorar a qualidade do workflow definido pelo pesquisador através de processos de inspeção. Terceiro,

ela permite que os resultados dos programas sejam armazenados automaticamente em *data warehouse*. Por fim, ela facilita a otimização do workflow através de processos que distribuem dados de entrada, fragmentam registros de bancos de dados, e unem ou intercalam os resultados parciais formando um resultado final.

Os processos envolvidos com a otimização são definidos na ontologia proposta nesta tese. A implementação de um SGABio, descrita no Capítulo 6, é capaz de utilizá-los para melhorar o workflow definido pelo pesquisador, o que não é feito pelos outros sistemas existentes. Por exemplo, [Cannataro et al., 2004] mostra um exemplo, no Proteus, de um workflow que paraleliza *n* processos BLAST, onde cada BLAST faz a comparação de um sub-conjunto de sequências de entrada. Em nossa proposta, esta otimização é feita automaticamente, ou seja, sem o conhecimento do pesquisador. No Proteus, o pesquisador deve ser capaz de definir todas as tarefas relacionadas a esta otimização, como a inclusão de um processo para realizar a divisão dos sub-conjuntos, a definição dos *n* processos BLAST, e o acréscimo de um processo para realizar a união dos resultados.

Quanto à alocação dos processos, dados e recursos aos processadores, na proposta desta tese, esta tarefa seria feita de forma automática pelo SGWBio. De fato, o SGWBio obtém, da ontologia, as propriedades dos processadores e seus relacionamentos com os processos e dados. Para cada processador existente, a ontologia informa a sua capacidade (ou seja, suas características fixas, como velocidade de CPU e tamanho de disco), os programas de Bioinformática instalados e os dados disponíveis. Com isso, o gerente de execução, conhecendo a ocupação do processador (ou seja, suas características em um determinado momento como espaço em disco disponível), pode escolher qual é o melhor processador para executar um determinado processo. Este tipo de tarefa não é feito pelos sistemas existentes. Por exemplo, a alocação dos processos aos processadores no grid do sistema Proteus é feita manualmente pelo pesquisador, embora ajudado por um sistema assistente. Ao transferir a responsabilidade desta tarefa para o pesquisador, corre-se o risco dele não ser capaz de definir seu workflow de forma coerente e otimizada. Ou seja, é possível que ocorram erros durante a definição do workflow, como a alocação de um processo a um processador onde ele não esteja instalado, ou a utilização desbalanceada de processadores.

Além disto, os sistemas existentes só permitem que o workflow seja otimizado *a priori*, enquanto a nossa proposta oferece otimização em dois passos: *a priori* e *a posteriori*. Na otimização *a posteriori*, o gerente de execução conhece a ocupação das máquinas e otimiza o workflow e a alocação dos processadores de acordo com esta ocupação. Desta forma, esta otimização permite que o sistema esteja preparado para atender um ou vários workflows ao mesmo tempo. Isto é importante porque os programas de Bioinformática possuem comumente alta complexidade de tempo e analisam um grande volume de dados, o que, consequentemente, pode tornar inviável a execução de vários workflows ao mesmo tempo. Sendo assim, a otimização *a posteriori* será mais adequada para a execução de um conjunto de workflows do que uma otimização exclusivamente *a priori*, como é feito pelos outros sistemas.

As otimizações propostas para os workflows em Bioinformática são feitas de acordo com o ambiente em que o SGWBio está implementado. O pipelining, a paralelização de processos e de cópias de processos através de distribuição de dados de entrada e da fragmentação de registros de bancos de dados, e as formas de implementação dos contêineres são estratégias de otimização propostas. O objetivo principal destas otimizações é a redução de espaço, tempo de execução do workflow e tempo de apresentação dos resultados intermediários do workflow ao pesquisador.

A tese incluiu ainda a proposta de uma linguagem para definição de workflows em Bioinformática, especificada através de um XML Schema. Embora existam diversas linguagens que poderiam ser utilizadas, todas teriam que ser adequadas para contemplar as nuances dos dados considerados em Bioinformática (como a leitura e geração de dados gradativa e não gradativa e o tamanho dos contêineres), que permitiram a aplicação das estratégias de execução e otimização propostas nesta tese. Desta forma, optou-se pela criação de uma linguagem simples, fácil de ser entendida, que não se compromete com a ontologia e que é adequada para definir todos os workflows nos ambientes considerados nesta tese. O comprometimento da linguagem com a ontologia implicaria na limitação da definição da ontologia, o que não estaria de acordo com o *framework* proposto para o SGWBio.

Por fim, a tese apresentou um protótipo construído instanciando-se o *framework* para o ambiente pessoal, mas que pode ser utilizado como base para os

outros ambientes. No protótipo foram desenvolvidos os módulos assistente, controlador, gerente de ontologia, gerente de otimização e gerente de execução.

Como a interação do sistema com o pesquisador não deve exigir que o pesquisador conheça características de ambiente, o módulo assistente é independente dos tipos de ambientes em que o sistema será implementado. Embora possa ser estendido para melhorar a iteração com o usuário, ele está implementado de forma a ser utilizado em qualquer ambiente.

As funcionalidades dos módulos controlador, gerente de ontologia e gerente de otimização são, em grande parte, independentes do ambiente. Consequentemente, estes módulos, embora possam ser estendidos, tornando-se mais robustos, podem ser utilizados em qualquer ambiente.

O módulo gerente de execução é o mais dependente do ambiente em que o SGWBio for instanciado. No protótipo, o gerente de execução simula os algoritmos definidos no Capítulo 6, considerando que não há limite de espaço em disco. Desta forma, todos os processos que fazem parte do *estágio_ideal* são disparados de forma concorrente. Entretanto, o projeto do gerente de execução foi feito permitindo sua extensão para ser utilizado em qualquer ambiente, de acordo com a discussão do Capítulo 6.

8.2 Trabalhos Futuros

É possível destacar vários trabalhos futuros a esta tese.

Primeiro, é interessante instanciar o *framework*, utilizando o protótipo já construído, para implementar e testar o SGWBio nos diferentes ambientes propostos segundo as estratégias de execução e otimização definidas no Capítulo 6.

Um estudo sobre as estimativas das taxas de consumo e produção dos programas de Bioinformática e dos tamanhos dos dados que estes programas lêem e produzem é muito válido, pois estas estimativas são utilizadas pelos algoritmos definidos no Capítulo 6 como base para as escolhas das otimizações mais adequadas e, consequentemente, são fundamentais para que as heurísticas definidas funcionem de forma adequada.

É ainda possível estender os algoritmos propostos no Capítulo 6 para contemplar outros tipos de otimizações como, por exemplo, otimização global e a

melhoria no gerenciamento de memória feito pelos programas de Bioinformática [Lemos, 2000a; Lemos, 2003b] já definidas no Capítulo 6, mas não utilizadas pelos algoritmos propostos. Estas extensões visam melhorar o tempo e o espaço utilizados para a execução dos workflows.

A ontologia também pode ser mais sofisticada, por exemplo, contemplando outras propriedades de qualidade para as classes de processos, de dados e dos processadores. Estas informações melhorarão ainda mais a definição e a execução otimizada dos workflows.

Outros requisitos levantados para o SGWBio e não enfatizados nesta tese também merecem ser estudados, como o monitoramento e o agendamento de workflows.

O sistema de gerência de dados em Bioinformática, que foi discutido de forma resumida no Capítulo 5, também pode ser estudado de forma mais detalhada estendendo os módulos definidos para o SGABio, visando a elaboração de uma ferramenta sofisticada que permita ao pesquisador analisar e interpretar de forma simples os dados gerados pelos workflows.