BLASTFYER: UMA FERRAMENTA DISTRIBUÍDA PARA ALINHAMENTO DE SEQUÊNCIAS BIOLÓGICAS

RENATA ZOTTIS JUNGES[1]; LÚCIO LEAL BASTOS[1]; MARILTON SANCHOTENE DE AGUIAR[1]

[1]*Universidade Federal de Pelotas – {rzjunges, llbastos, marilton}@inf.ufpel.edu.br*

1. INTRODUÇÃO

A ferramenta *Basic Local Alignment Search Tool* (BLAST) busca regiões de similaridade entre sequências biológicas (ALTSCHUL, 1990), comparando sequências de nucleotídeos ou de proteínas contra bases de dados e calculando a significância estatística entre os pares (NCBI, 2017). Além disso, ele pode ser usado para inferir relações funcionais e evolutivas entre as sequências, auxiliando na identificação de membros de famílias gênicas (PORTAL EDUCAÇÃO, 2013).

Por essa razão ele é um dos programas de alinhamento mais utilizados cientificamente sendo de extrema importância para a bioinformática. Em contrapartida, caso seja necessário comparar um grande número de sequências biológicas com diversos bancos de dados ocorrerá um aumento considerável no tempo de processamento pois a busca por regiões de similaridade é feita sequencialmente.

Atualmente, no ambiente acadêmico, é comum existir laboratórios com muitos computadores, ociosos por muito tempo. Nesse contexto, pensou-se numa estratégia para minimizar o tempo de processamento do BLAST quando existe uma quantidade considerável de buscas de similaridade e maximizar a utilização dos computadores que não estão em uso.

Neste sentido, este trabalho visa apresentar a ferramenta BLASTFYER que está em desenvolvimento no Laboratório de Sistemas Inteligentes da Computação da UFPel. Esse *software* realiza a distribuição, em uma rede de computadores, das tarefas executadas sequencialmente pelo BLAST, minimizando, dessa forma, o tempo total de processamento.

2. METODOLOGIA

O BLASTFYER foi desenvolvido em um modelo cliente-servidor para realizar a distribuição de atividades executadas pelo BLAST. Nesse tipo de modelo, o servidor refere-se a um programa em execução em um computador interligado em rede, aceitando pedidos de programas em execução em outros computadores - conhecidos como clientes (COULOURIS et al, 2013).

No *software* desenvolvido, um dos computadores da rede opera como servidor, contendo a parte da ferramenta que prepara e distribui as tarefas; e, os demais computadores operam como clientes, executando a ferramenta BLAST sobre cada conjunto de dados separadamente. Quando um cliente encerra a atividade que lhe foi solicitada, este se mantém disponível para receber outra atividade proveniente do servidor, mantendo-se neste ciclo até que o servidor não possua mais tarefas disponíveis para processamento.

Para o desenvolvimento do BLASTFYER utilizou-se a linguagem de programação *Python 3* juntamente com o *sqlite 3*, um sistema gerenciador de banco de dados da aplicação. Para executar o *software* utiliza-se o terminal (em sistemas operações da distribuição *Linux*) ou *prompt* de comando (no sistema operacional *Windows*).

O computador designado como servidor, executa o BLAST com argumentos parametrizados de forma refinada, definidos pelos autores a partir de testes preliminares (argumentos que são obrigatórios na utilização original do BLAST permanecem obrigatórios na execução do BLASTFYER). Por outro lado, os computadores que serão utilizados como clientes informam ao servidor através da linha de comando qual é o endereço IP do computador que está executando a aplicação, no caso, o servidor.

Além disso, para que seja possível a execução do BLASTFYER, é necessário ter acesso a um servidor *File Transfer Protocol (*FTP) onde estão armazenadas as sequências biológicas e os banco de dados que serão utilizados. FTP é um protocolo de transferência de arquivos, que utiliza a conexão TCP, para transferir arquivos de/para um servidor remoto (KUROSE, 2013).

Para mostrar como devem ser as linhas de comando, Na Figura 1 é apresentado um exemplo referente à configuração do servidor (na primeira linha) e, outro, referente aos clientes (segunda linha).

```
> python3 server.py -ftp user:pass@255.255.255.255
  -query pasta/sequencias/a*.fasta
  -db pasta/db/u*.fasta
  -out pasta/resultados -outfmt 5
> python3 client.py -host 255.255.255.200
```

Figura 1. Exemplo de Parametrização do Servidor e do Cliente

Em relação a aplicação servidor, é indicado o endereço do servidor FTP com o usuário e senha que devem ser submetidos pelo cliente, os argumentos `query`, `db` e `out` se referem aos arquivos de sequências biológicas, bancos de dados e os arquivos de resultados, respectivamente. Todos estes argumentos devem ser escritos utilizando o caminho absoluto a partir do diretório raíz do servidor FTP.

As sequências biológicas e os bancos de dados estão no servidor FTP para que os clientes possam fazer o *download* destes e executarem o comando BLAST; e, após a execução é gerado um arquivo de resultado final e é feito o *upload* deste arquivo para o servidor FTP.

Somente a partir da linha indicada pelo servidor é possível determinar quantas tarefas deverão ser distribuídas e executadas. Supondo que no caminho `pasta/sequencias` existam 300 arquivos que começam com a letra ‘a’ e extensão ‘.fasta’ e no caminho `pasta/db` existam 5 arquivos que iniciem com a letra ‘u’ e extensão ‘.fasta’, então o servidor organiza 1500 tarefas, ou seja, todas as sequências biológicas contra todos os banco de dados.

A partir do momento que é executada a linha de comando no servidor, ele aguardará que os clientes se conectem através da rede. Para cada novo cliente conectado é aberta uma *thread* que será de uso exclusivo para cada cliente e é através dela que o servidor consegue comunicar-se com o cliente e receber mensagens informando se houve sucesso na execução e no *upload* do arquivo que contém o resultado do processamento. Desse modo, o servidor tem controle

dos computadores online no momento e da ociosidade de cada um.

Além disso, quando um cliente informa que encerrou uma atividade, este entra na lista de computadores ociosos e então o servidor lhe envia uma nova tarefa. Apenas quando todas as atividades do servidor são processadas e todos os arquivos de resultados estão no servidor FTP é que o programa deve encerrar. Assim, o servidor envia uma mensagem para todos os clientes conectados informando que o processamento total já foi realizado e que eles devem se desconectar; e, após isso, o servidor é automaticamente fechado.

3. RESULTADOS E DISCUSSÃO

O BLASTFYER foi testado, preliminarmente, utilizando 99 arquivos de sequências biológicas e 2 bancos de dados, resultando em 198 combinações distintas. Os testes foram realizados de duas formas diferentes, para expressar a diferença de velocidade de execução em uma rede cliente-servidor para uma execução local: execução distribuída em uma rede com 19 clientes, com *download* dos arquivos e *upload* dos resultados via FTP; e, execução local (apenas 1 cliente), com *download* dos arquivos e *upload* dos resultados via FTP.

Em ambos os testes os tempos de *download* dos arquivos e *upload* dos resultados foram desconsiderados, restando apenas o tempo de processamento da ferramenta. Além disso, os computadores usados no teste foram modelos HP 6300 MT com processador Intel Core i3-3240, 4GB de memória RAM e Sistema Operacional Ubuntu 16.04.

Cada teste foi executado 15 vezes e foram comparadas as médias dos tempos de execução. A Figura 2 ilustra os resultados obtidos destas execuções.

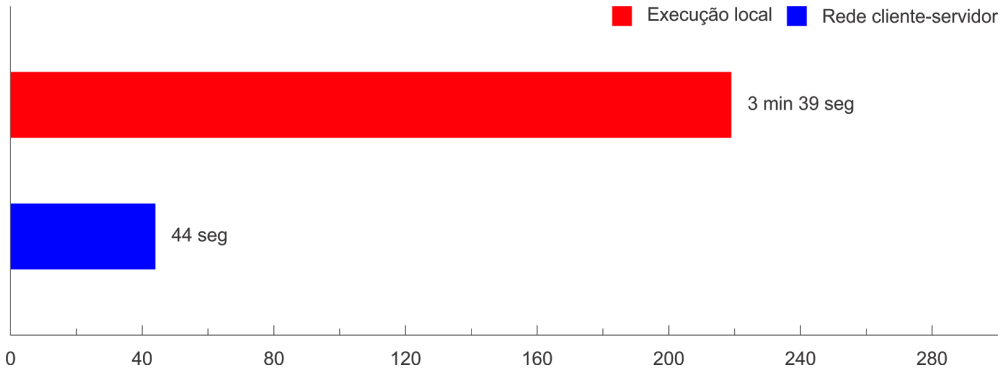

Figura 2. Gráfico comparativo dos tempos de execução da ferramenta BLASTFYER utilizando uma rede cliente-servidor contra uma execução local.

Na execução local, a média do tempo de execução foi de 3 minutos e 39 segundos. Porém na execução através de rede cliente-servidor obteve-se tempo médio de 44 segundos, o que representa 20,09% do tempo necessário para executar a ferramenta localmente.

4. CONCLUSÕES

O BLASTFYER proporciona uma redução no tempo de processamento de análise de sequências biológicas pela ferramenta BLAST, utilizando-se para isso uma rede cliente-servidor, com computadores heterogêneos, em uma rede local ou dispersos geograficamente, de forma transparente para o usuário da ferramenta. O software continua a receber melhorias e correções e o seu registro foi realizado junto ao INPI, sob o número de processo BR 51 2017 000935-0.

5. REFERÊNCIAS BIBLIOGRÁFICAS

ALTSCHUL, S.F.; GISH, W.; MILLER, W.; MYERS, E.W.; LIPMAN, D.J. Basic Local Alignment Search Tool. **Journal of Molecular Biology**, v.215, n.3, p.403-410, 1990.

COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T.; BLAIR, G. **Sistemas Distribuídos: Conceitos e Projeto**. Bookman Editora, 2013.

KUROSE, J.F.; ROSS K.W. **Redes de Computadores e a Internet.** São Paulo: Person, 2013. 6v.

NCBI. **Basic Local Alignment Search Tool.** Bethesda MD, USA. Acessado em 25 set. 2017. Online. Disponível em: https://blast.ncbi.nlm.nih.gov/Blast.cgi

PORTAL EDUCAÇÃO. **O BLAST e a bioinformática.** Campo Grande-MS, 28 fev. 2013. Acessado em 25 set. 2017. Online. Disponível em: https://www.portaleducacao.com.br/conteudo/artigos/idiomas/o-blast-e-a-bioinformatica/36399