

## 5.

### **Avaliação de Ferramentas BLAST em Ambientes Distribuídos**

No presente capítulo são examinados os fatores a considerar quando se deseja efetuar uma avaliação entre ferramentas BLAST voltadas a um ambiente distribuído. Em seguida, são apresentados alguns dos testes realizados, para avaliar e comparar as ferramentas BLAST, em particular o balaBLAST.

Para efetuar a avaliação e a comparação das ferramentas BLAST, foram utilizados alguns dos conceitos apresentados no Capítulo 2 e as ações propostas no roteiro constante do Anexo 3 da presente dissertação.

#### 5.1

##### **Processo de Seleção de Ferramentas para Avaliação.**

Como foi mostrado no Capítulo 2, o processo de avaliação de uma ferramenta de *software* deve levar em consideração fatores tais como: desempenho; abrangência; funcionalidade, usabilidade, entre outros. No presente capítulo estamos interessados especificamente em avaliar e comparar o desempenho de ferramentas BLAST em um ambiente distribuído.

No Capítulo 3 foram abordadas algumas ferramentas desenvolvidas para a execução do BLAST em paralelo. Considerando os fatores discutidos no Capítulo 2, pode não ser possível efetuar uma avaliação incluindo todas as ferramentas.

A execução de uma ferramenta de *software* depende fundamentalmente do ambiente computacional existente em uma organização. Não existe possibilidade de se tentar efetuar a avaliação de desempenho de ferramentas que não possam ser executadas em função do *hardware* e *software* existentes em uma instalação. Por

exemplo, o fato de uma ferramenta rodar em um ambiente de *cluster*, não significa que possa ser executada em um ambiente de *grid*.

Nesta dissertação, seguindo roteiro apresentado no Anexo 3, inicialmente foram identificadas quais as ferramentas que poderiam ser utilizadas para efetuar a avaliação, além do balaBLAST. Observando os recursos existentes na PUC-Rio e na UFF, mostrados no Capítulo 4, e as ferramentas descritas no Capítulo 3, foi possível constatar que apenas o balaBLAST e o mpiBLAST podiam ser executados nestes ambientes, em virtude de:

- a) A ferramenta desenvolvida pelo projeto *Bridges* utilizar o Globus Toolkit 3 como *middleware* básico, assim como o Package BLAST;
- b) O Grid-Blast utilizar comandos específicos ao NorduGrid, que não são compatíveis com os ambientes disponíveis para os testes;
- c) O W.nd Blast utilizar o Windows como sistema operacional;
- d) A versão do Squid disponível para download executar apenas em ambiente Windows;
- e) O HGBS consistir em uma implementação baseada em Hardware, não disponível nos ambientes considerados para os testes.

Em seguida foi efetuada uma análise complementar, na qual foram considerados outros fatores citados na Seção 2.1, a partir da qual foi verificado que:

- a) O mpiBLAST é uma das ferramentas BLAST para ambientes distribuídos mais utilizadas pelos biólogos e bioinformatas e, portanto, a sua correção ser atestada pelo seu uso no meio científico (acurácia).
- b) Tanto o mpiBLAST quanto o balaBLAST oferecem facilidades de comparação entre seqüências de nucleotídeos e proteínas para identificar homologias e predições de funções de genes (funcionalidade);
- c) Ambas as ferramentas produzem os mesmos resultados todas as vezes que são utilizados, considerando a utilização de mesmos parâmetros em sua execução (acurácia);

- d) Os resultados obtidos por ambas as ferramentas são corretos, mesmo porque chamam um mesmo programa “blastall” do NCBI BLAST, utilizando os mesmos parâmetros, programa este bastante utilizado na comunidade científica;
- e) Ambas as ferramentas são executadas a partir de uma linha de comando, na qual os parâmetros de execução desejados são discriminados (usabilidade);
- f) As duas ferramentas não oferecem obstáculos, quanto ao seu entendimento, aos seus potenciais usuários (facilidade);
- g) Apesar do mpiBLAST exigir uma máquina a mais do que o balaBLAST para a sua execução, tal fato não é proibitivo à sua execução na instalação existente (adequabilidade).

Portanto ambas as ferramentas, o mpiBLAST e o balaBLAST, podem então ser consideradas para a avaliação.

## 5.2

### **GradeBLAST**

Considerando o interesse em medir o desempenho de ferramentas que utilizam a replicação da base de dados como estratégia de paralelismo frente àquelas que utilizam estratégias de fragmentação, assim como o desempenho obtido por ferramentas escritas em linguagens de script, como é o caso do Perl, foi desenvolvida uma ferramenta, escrita em *bash*, capaz de executar o BLAST em um ambiente distribuído. A ferramenta assim desenvolvida foi denominada, gradeBLAST, e constitui a terceira ferramenta a ser considerada para avaliação de desempenho.

O gradeBLAST possui como características: a utilização de comandos do Globus Toolkit para copiar e executar programas em nós remotos; é composta por apenas um programa que roda no nó mestre e que faz o controle das execuções nos diversos nós, ao contrário de outras ferramentas BLAST baseadas em linguagem de script mostradas no Capítulo 3; possui rotinas voltadas à

recuperação de falhas; e produz um relatório consolidado das execuções do NCBI BLAST nos diversos nós. Uma descrição mais detalhada sobre o gradeBLAST pode ser obtida no Anexo 1, e o código fonte em um CD que acompanha a presente dissertação.

### 5.3

#### **Avaliação e Comparação de Desempenho**

Uma vez selecionadas as ferramentas BLAST, foi possível então proceder à avaliação de desempenho entre elas. Cabe aqui mencionar que não existe uma especificação padrão de testes para se efetuar um *benchmark* de ferramentas BLAST em paralelo, e por este motivo foi necessária a definição de testes para efetuar a avaliação e a comparação entre essas ferramentas.

A principal base de dados biológica considerada na elaboração dos testes foi a base de proteínas não redundante NR, que possuía aproximadamente 4 GB e 7 milhões de registros no início de 2009. Outra base utilizada para os testes foi a Swisprot, também de proteínas, mas possuindo um tamanho bem menor, cerca de 96 MB, ou seja, aproximadamente 2,4% da base NR.

Como principal métrica utilizada para efetuar a comparação foi adotado o “Tempo de Execução”, medido em horas, minutos e segundos. A métrica “Número de E/S”, também utilizada, teve por objetivo identificar as possíveis causas para diferentes tempos de execução encontrados nos testes realizados nas ferramentas selecionadas. É importante observar que ambas as métricas, consideradas na avaliação das ferramentas BLAST analisadas, atenderam aos critérios estabelecidos na Seção 2.1 e, portanto, permitiram:

- a) que fossem efetuadas medidas corretas, com significados concretos;
- b) descrever realmente o que se estava medindo;
- c) entender facilmente o que se estava analisando;
- d) capturar dados realmente importantes à análise; e
- e) não efetuar medições redundantes.

Quatro tipos de arquivos de consultas foram montados com seqüências retiradas da própria base de dados NR: 500 seqüências e aproximadamente 500 aminoácidos cada; 500 seqüências e aproximadamente 1000 aminoácidos cada; 1000 seqüências e aproximadamente 500 aminoácidos cada; e 1000 seqüências e 1000 aminoácidos cada.

Assim, para os quatro tipos de consultas definidos, os tempos de execução das ferramentas, utilizando parâmetros *default* de cada uma delas, foram medidos para os ambientes de *cluster* e/ou de *grid* existentes. Para cada tipo de consulta foi utilizado um mesmo conjunto de equipamentos ou, pelo menos, equipamentos de mesmo porte. Adicionalmente, foram coletados dados relativos às quantidades de E/S efetuados, utilizados para auxiliar à análise dos resultados obtidos.

O tamanho das tarefas a serem processadas em cada um dos nós, durante a execução do balaBLAST, deve ser fixado como um múltiplo do número de nós trabalhadores a serem considerados para um *cluster* ou para um *grid*. Assim, considerando, por exemplo, 8 nós trabalhadores, podem ser estipulados, como tamanho das tarefas, valores tais como 8, 16, 24, 32, etc.. A utilização de um tamanho muito grande para as tarefas pode implicar na não utilização do algoritmo de balanceamento de carga pelo balaBLAST, durante a sua execução, tornando a ferramenta ineficiente. Por outro lado, a adoção de um tamanho menor para as tarefas pode implicar na perda de eficiência da ferramenta, motivada pelo número excessivo de movimentações de dados entre o nó mestre e os nós trabalhadores do *cluster* ou do *grid*.

Nas avaliações efetuadas durante o presente trabalho, optou-se por um tamanho de tarefas igual a 24, um valor intermediário, considerando o número de tarefas a serem analisadas pela ferramenta em cada um dos nós (aproximadamente 21, por nó, para consultas com 500 seqüências, e 42, para consultas com 1000 seqüências), e que proporcionou uma boa margem de distribuição de tarefas pelo balaBLAST entre os nós trabalhadores, durante os testes realizados. Por uma questão de coerência, o mesmo tamanho de tarefas foi adotado também para os testes realizados com o gradeBLAST.

Por não oferecer a opção de utilização de tarefas para sua execução, tal como definido para o balaBLAST, foram adotadas as configurações padrões do mpiBLAST durante os testes efetuados.

Para efetuar as avaliações foram utilizados os equipamentos discriminados no Capítulo 4, Tabela 4.1, pertencentes à PUC-Rio e à UFF, sendo normalmente utilizados nos testes os equipamentos Intel Duo Core com 2.16 GHz com 1GB de RAM, pertencentes à PUC-Rio e os equipamentos Intel Quad Core 2.4 GHz com 3GB de RAM, pertencentes à UFF.

As métricas de E/S, tanto para leituras em disco, quanto para gravações foram obtidas através de uma ferramenta, desenvolvida com este propósito, capaz de recuperar os dados necessários à sua obtenção, acessando os arquivos virtuais mantidos pelo sistema operacional Linux no diretório “/proc”.

Para efetuar a avaliação e a comparação das ferramentas foram considerados os seguintes testes, discriminados no Anexo 5:

- a) Base NR, 8 equipamentos da PUC-Rio, 8 fragmentos, consulta 500 seqüências e 500 aminoácidos:

Teste inicial para se ter uma base de comparação;

- b) Base NR, 8 equipamentos da UFF, utilizando grid, 8 fragmentos, consulta com 500 seqüências e 500 aminoácidos:

Verificação do comportamento das ferramentas em um grid, e comparação com o desempenho obtido em um ambiente de cluster;

- c) Base NR, 4 equipamentos da PUC-Rio, 4 e 8 fragmentos, consulta com 500 seqüências e 500 aminoácidos:

Análise do comportamento das ferramentas em um cluster, considerando a diminuição do número de máquinas, e o fato dos fragmentos não caberem na memória principal. No caso do balaBLAST e do mpiBLAST, a análise do comportamento quando da existência de mais de um fragmento utilizável para comparação por máquina. Comparação com execuções utilizando 8 equipamentos;

- d) Base NR, 8 equipamentos da PUC-Rio, 8 fragmentos, consulta com 500 seqüências e 1000 aminoácidos:

Análise do comportamento das ferramentas considerando um acréscimo do número de aminoácidos por seqüência de uma consulta, mantendo o mesmo número de seqüências;

- e) Base NR, 8 equipamentos da PUC-Rio, 8 fragmentos, consulta com 1000 seqüências e 500 aminoácidos por seqüência da consulta:

Análise do comportamento das ferramentas considerando um acréscimo do número de seqüências de uma consulta, mantendo o mesmo número de aminoácidos por seqüência;

- f) Base NR, 8 equipamentos da PUC-Rio, 8 fragmentos, consultas com 1000 seqüências com 1000 aminoácidos:

Análise do desempenho das ferramentas quando as seqüências de consulta são duplicadas assim como o número de aminoácidos, em relação à consulta com 500 bases e 500 aminoácidos;

- g) Base NR, 8 equipamentos da UFF, utilizando grid, 8 fragmentos, consulta com 1000 seqüências com 1000 aminoácidos:

Análise da conduta das ferramentas em um ambiente de grid, considerando consulta com o dobro das seqüências e de aminoácidos, em relação à execução anterior no ambiente de grid. Comparação com o desempenho resultante de execuções anteriores.

- h) Base NR, 4 equipamentos da PUC-Rio, 4 fragmentos, consulta com 1000 seqüências e 1000 aminoácidos:

Análise do comportamento das ferramentas utilizando 4 máquinas, considerando um aumento do número de seqüências e aminoácidos por seqüência, da mesma maneira que no teste anterior;

- i) Base Swissprot, 4 equipamentos da PUC-Rio, 4 fragmentos, consulta com 500 seqüências e 500 aminoácidos e 1000 seqüências e 1000 aminoácidos:

Verificação do comportamento das ferramentas considerando uma base de dados de muito menor porte que a base NR. Comparação de desempenho entre as ferramentas em análise.

Com o objetivo de não distorcer os resultados obtidos pelo fato dos fragmentos da base de dados serem movimentados pelo mpiBLAST para os nós trabalhadores em tempo de execução, os seguintes procedimentos foram adotados: fazer constar no arquivo de configurações do mpiBLAST “.ncbirc”, que o diretório local e compartilhado eram os mesmos; e, antes de cada teste, movimentar todos os fragmentos da base de dados biológica, a serem utilizados no processamento das seqüências de consulta, para os diretórios locais definidos em cada um dos nós trabalhadores. Tais procedimentos induziram a que o mpiBLAST não efetuasse movimentações de fragmentos entre os nós durante a sua execução, fazendo com que ele se comportasse exatamente como o balaBLAST.

Uma observação importante diz respeito à forma com que os testes foram executados nos equipamentos da UFF. Normalmente as aplicações de *grid* que são executadas na UFF utilizam uma área em disco compartilhada. Para evitar possíveis distorções em razão da utilização dessa área, foi acertado com a equipe de suporte UFF que os testes fossem executados utilizando-se uma área em disco temporária de acesso local, ou seja, sem a necessidade de usar áreas em disco compartilhadas com outras aplicações clientes do *grid*.

### 5.3.1

#### **Teste utilizando 8 Equipamentos do Cluster da PUC-Rio para Consultas com 500 Seqüências e Aproximadamente 500 Aminoácidos**

Inicialmente foi processada uma consulta contendo 500 seqüências com aproximadamente 500 aminoácidos cada, utilizando oito nós do *cluster* da PUC, sendo que para o gradeBLAST e o balaBLAST foram considerados 8 fragmentos, e 24 seqüências por tarefa.

Embora o gradeBLAST seja uma ferramenta desenvolvida especialmente para um ambiente de *grid*, ela pode funcionar em um *cluster*, desde que o Globus Toolkit tenha sido instalado, e que o usuário possua um certificado que permita efetuar a autenticação para acesso.



A Figura 5.1 destaca o menor tempo de execução para o balaBLAST, cerca de 1 hora e 37 minutos, seguido do mpiBLAST e gradeBLAST, com um tempo superior a 2 horas. A Figura 5.2 permite comparar a quantidade de leituras em disco efetuadas pelas três ferramentas, e mostra que o balaBLAST teve o melhor desempenho entre elas. Tal fato justifica que o balaBLAST tenha sido o mais rápido ao processar as seqüências de consulta submetidas durante o teste.

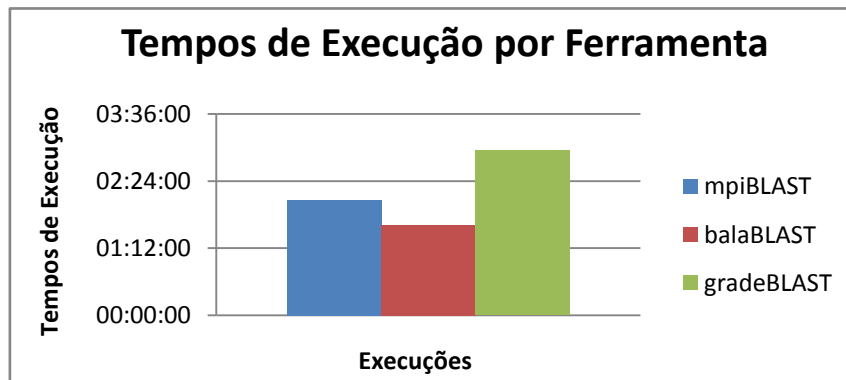


Figura 5.1: *Tempos de execução para as ferramentas*

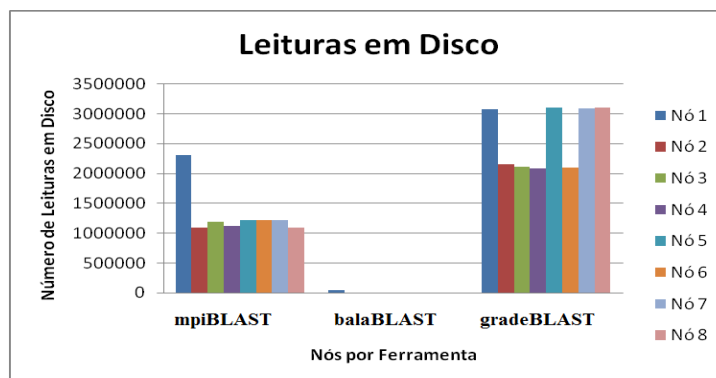


Figura 5.2: *Comparação da quantidade de leituras em disco efetuadas*

O resultado obtido pelo balaBLAST tem como explicação a estratégia de E/S utilizada, que permite reduzir o tempo de execução da ferramenta através à carga dos fragmentos da base de dados antes da execução do programa “blastall” na memória RAM, permitindo assim que haja uma única leitura, e, portanto, uma queda significativa na quantidade de leituras necessárias à execução.

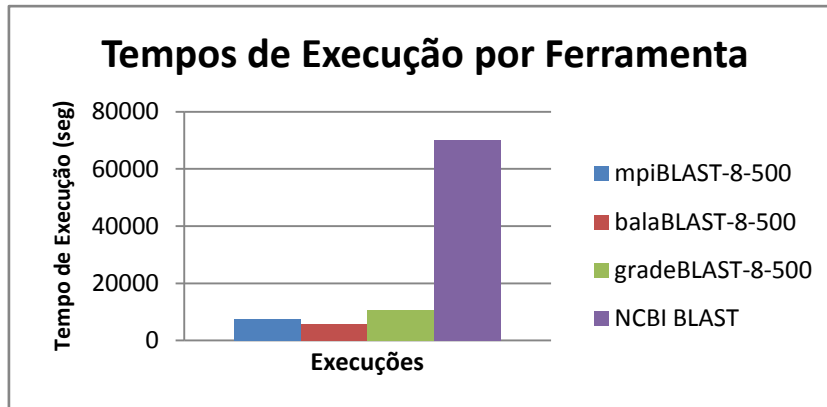


Figura 5.3: *Tempos de execução das ferramentas (segundos) em relação ao NCBI BLAST.*

É importante observar o ganho que se obteve através da execução do BLAST utilizando-se uma ferramenta que permite a sua execução em um ambiente distribuído. A Figura 5.3 mostra que houve uma aceleração (*speedup*) de aproximadamente 6,5 a 12 vezes, dependendo da ferramenta utilizada, ou ainda, um ganho de eficiência de aproximadamente 1,5 por máquina, considerando o tempo de execução do balaBLAST, em relação ao NCBI BLAST, que foi executado em apenas uma máquina. Tal fato justifica a utilização de ferramentas BLAST em paralelo.

### 5.3.2

#### Teste no Grid utilizando 8 Equipamentos da UFF para Consultas com 500 Seqüências e Aproximadamente 500 Aminoácidos

A mesma consulta da seção anterior foi efetuada utilizando um ambiente de *grid*, com duas máquinas da PUC-Rio, do tipo 1, e oito máquinas da UFF, no caso do mpiBLAST, e uma máquina da PUC-Rio e oito máquinas da UFF, para o balaBLAST e o gradeBLAST. Infelizmente, em virtude de problemas na UFF, uma das máquinas utilizadas era do tipo 4 (1GB de RAM, 1GB de disco, e um processador Intel Pentium 4 de 3.2 GHz), enquanto as demais eram do tipo 3.

Como pode ser observado na Figura 5.4, o resultado foi também favorável ao balaBLAST, com um tempo de execução 1 hora e 57 minutos, contra 2 horas e 24 minutos do mpiBLAST e 2 horas e 52 minutos do gradeBLAST. Embora os

tempos de execução do mpiBLAST e do balaBLAST tenham sido superiores à execução anterior na PUC-RIO, pode-se notar que houve uma melhora, da ordem de 0,02% no desempenho do gradeBLAST, que utiliza uma estratégia replicada. Tal melhora foi resultado da maior disponibilidade de memória RAM nos equipamentos da UFF em que o teste foi executado, que possuem 3GB, à exceção de um dos equipamentos conforme explicado anteriormente.

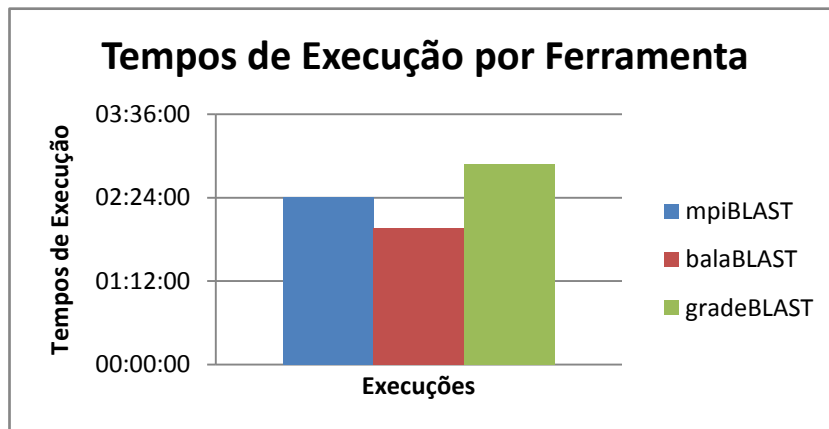


Figura 5.4: *Tempos de execução para cada ferramenta.*

Posteriormente, um teste adicional foi efetuado, no sentido de verificar o ganho do gradeBLAST ao substituir a máquina mais lenta, utilizada no teste anterior, por outra máquina, com memória RAM de 3GB, mantidas as demais, e o resultado mostrou que o tempo de execução do gradeBLAST, em relação aos resultados anteriores obtidos pelo balaBLAST e mpiBLAST, diminuiu mais ainda, com um tempo de execução de 2 horas e 25 minutos, praticamente igualando o desempenho do mpiBLAST, conforme mostra a Figura 5.5. Isto confirma o fato que o desempenho de ferramentas que utilizam estratégia replicada é dependente da quantidade de memória RAM disponível.

A Figura 5.6 permite comparar os tempos de execução obtidos nos dois testes acima. Percebe-se que, embora os equipamentos utilizados na UFF tenham maior capacidade de processamento, os tempos de execução obtidos, quando utilizando o *grid*, foram piores que aqueles obtidos quando do processamento na PUC-Rio, à exceção do gradeBLAST, como foi citado anteriormente. Existem duas explicações para o fato: a primeira, evidenciada na Figura 5.7, é que os tempos de comunicação entre o equipamento utilizado como mestre durante a

execução do balaBLAST, situado na PUC-Rio, e os demais equipamentos localizados na UFF, utilizados no teste, foram superiores em 154%, na sua totalidade, àqueles entre o mestre e os equipamentos locais pertencentes ao *cluster* da PUC-Rio; a segunda causa está relacionada ao fato de que em virtude de um *grid* não constituir um ambiente totalmente controlado, poderiam existir outras aplicações sendo executadas, concorrendo com a execução do BLAST, no momento em que os testes estavam sendo executados.

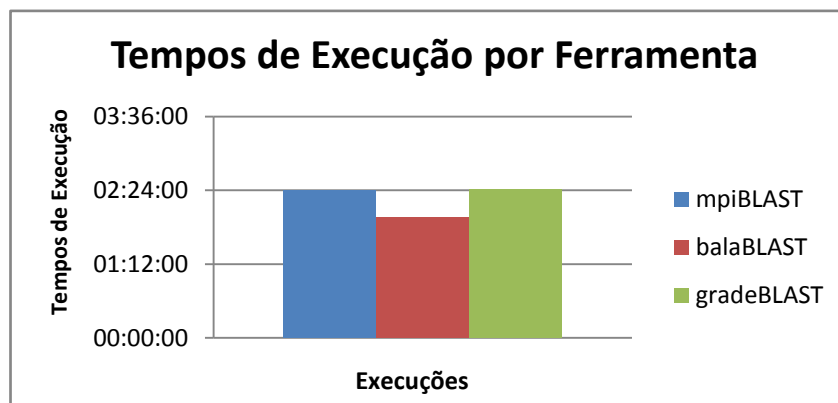


Figura 5.5: *Teste idêntico ao referenciado na Figura 5.4, porém somente com o gradeBLAST sendo executado com uma das máquinas com maior capacidade de processamento.*

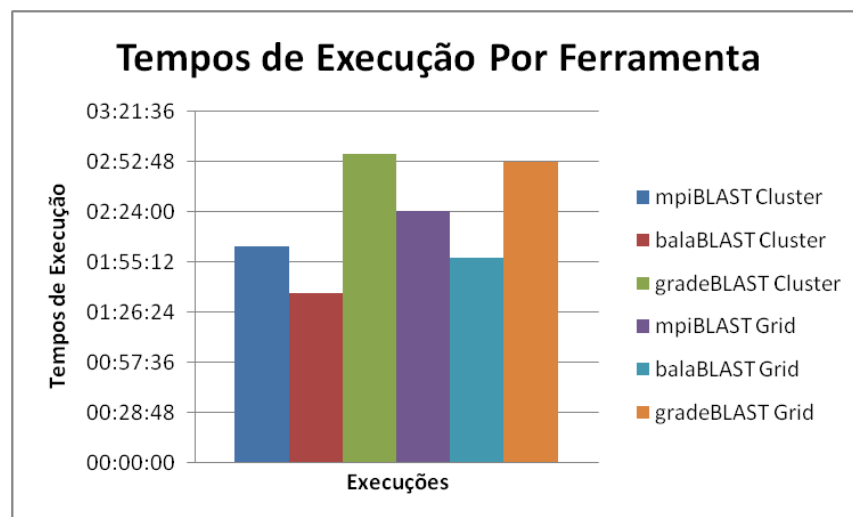


Figura 5.6: *Comparação dos tempos de execução das ferramentas na PUC-Rio e na UFF.*

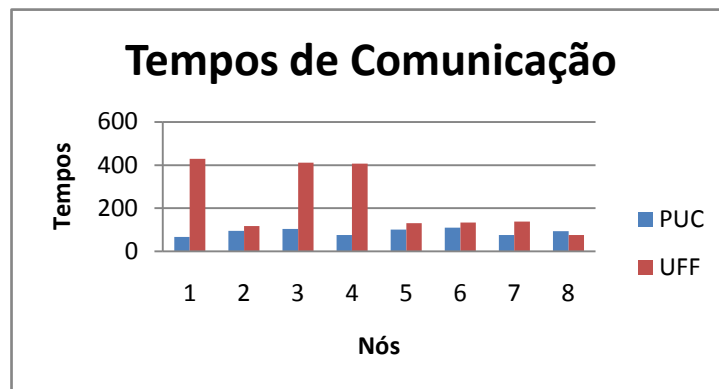


Figura 5.7: *Tempos de comunicação entre o mestre situado na PUC-Rio e os demais equipamentos durante o teste*

### 5.3.3

#### **Teste utilizando 4 Equipamentos do Cluster da PUC-Rio para Consultas com 500 Seqüências e Aproximadamente 500 Aminoácidos**

A Figura 5.8 mostra o tempo de execução de cada uma das estratégias utilizando apenas 4 equipamentos pertencentes ao *cluster* da PUC-Rio, e quatro fragmentos da base de dados. O resultado obtido, favorável ao balaBLAST, com um tempo de execução de 3 horas e 16 minutos contra 3 horas e 42 minutos do mpiBLAST e 5 horas e 10 minutos do gradeBLAST, demonstrou um considerável aumento no tempo de execução das três ferramentas. No caso do balaBLAST o tempo de execução foi duplicado, ou seja, caso fossem utilizadas 8 máquinas, ao invés, seria obtida uma aceleração de 2 vezes, com um ganho em termos de eficiência de 0,25 por equipamento. A Figura 5.9 que permite uma comparação visual entre os tempos obtidos pelas ferramentas quando utilizando 4 e 8 máquinas. Tal fato, no entanto, já era esperado, considerando que foram utilizados menos equipamentos e fragmentos da base de dados.

No caso das estratégias fragmentadas o incremento no tempo de execução ocorreu em virtude do acréscimo do tamanho de cada fragmento, que teve por consequência a necessidade de maior paginação da memória, assim como da

diminuição do número de processadores. Quanto à estratégia replicada, o acréscimo foi resultado, basicamente, da redução do número de processadores.

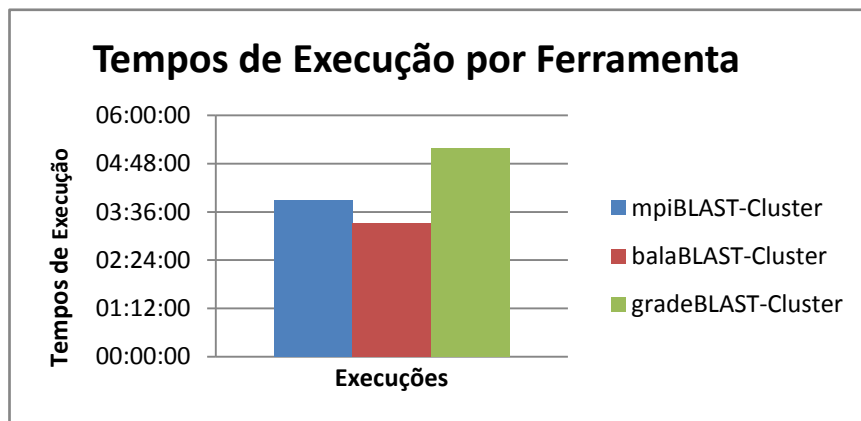


Figura 5.8: Tempos de execução por ferramenta para 4 máquinas e 4 fragmentos.

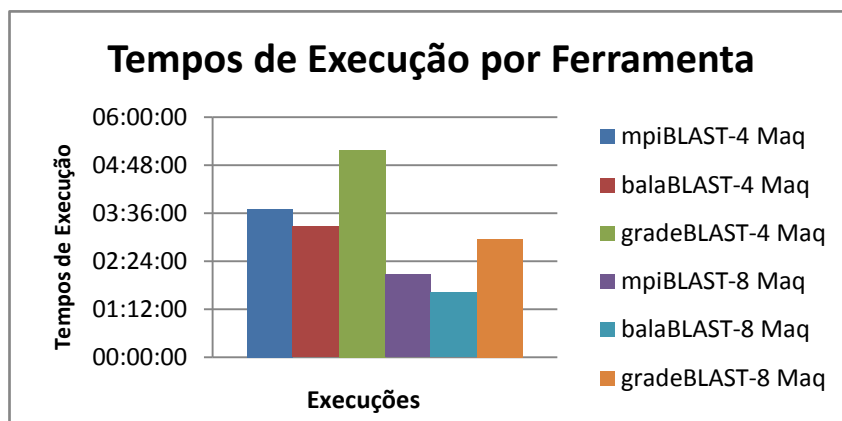


Figura 5.9: Comparação de resultados com 4 e 8 máquinas

A Figura 5.10 permite verificar as atividades de E/S, relativas a leituras em disco, e mostra que a quantidade de leituras efetuadas pelo balaBLAST foi bem menor que nas demais ferramentas, correspondendo a cerca de 20% das leituras efetuadas pelo mpiBLAST e de 4% das efetuadas pelo gradeBLAST, o que justifica o melhor resultado que obteve.

A Figura 5.11 apresenta os tempos de execução quando as ferramentas mpiBLAST e balaBLAST são rodadas com 4 máquinas, porém com 8 fragmentos. O balaBLAST apresentou um melhor desempenho com um tempo de execução de

3 horas e 16 minutos contra 4 horas e 13 minutos do mpiBLAST, apresentando um ganho de aproximadamente 22%.

A Figura 5.12 apresenta uma comparação entre execuções considerando 4 ou 8 máquinas e 4 ou 8 fragmentos. A Figura mostra que não houve uma grande variação em termos de tempo de execução, considerando execuções com 4 máquinas e 4 fragmentos, e 4 máquinas e oito fragmentos, sendo que para o balaBLAST houve um acréscimo de 2,5%, embora existisse uma atividade percentualmente bem maior de leituras em disco, da ordem de 68%, quando executando em 4 máquinas com 8 fragmentos, como mostram as Figuras 5.13 e 5.14.

O fato, citado no parágrafo anterior, induz a idéia de que o programa “blastall”, para cada tarefa, foi rodado duas vezes, ou seja, primeiro considerando um dos fragmentos primários, e logo após considerando o outro, ou seja, aproveitando ao máximo possível o fato de um dos fragmentos estarem em memória, o que realmente corresponde à estratégia utilizada para otimizar os acessos a disco. Tal fato explica o resultado obtido pelo balaBLAST, mostrado na Figura 5.11.

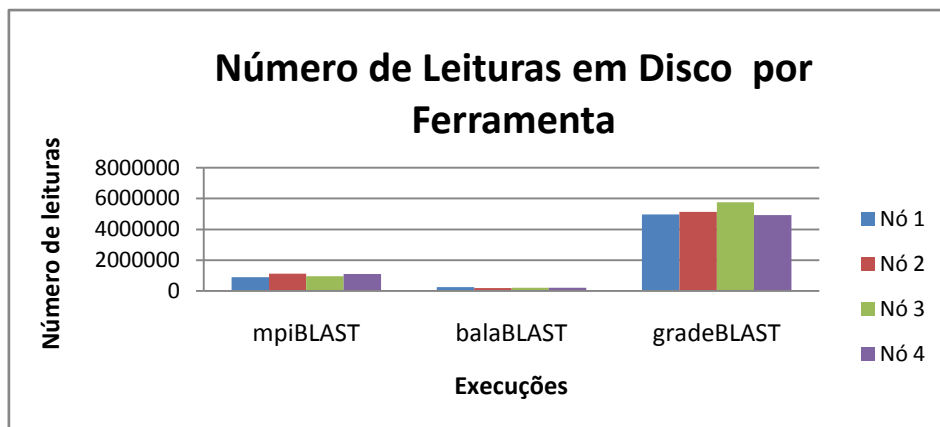


Figura 5.10: Leituras em disco efetuadas pelas ferramentas, para 4 máquinas e 4 fragmentos.

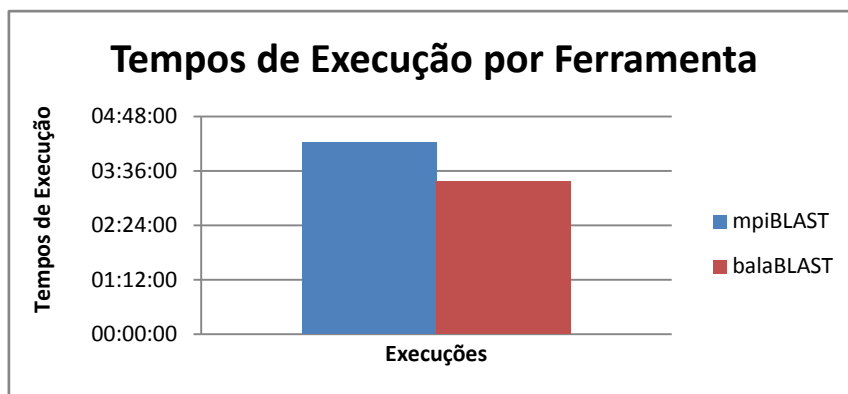


Figura 5.11: *Tempos de Execução, ao utilizar 4 máquinas e 8 fragmentos.*

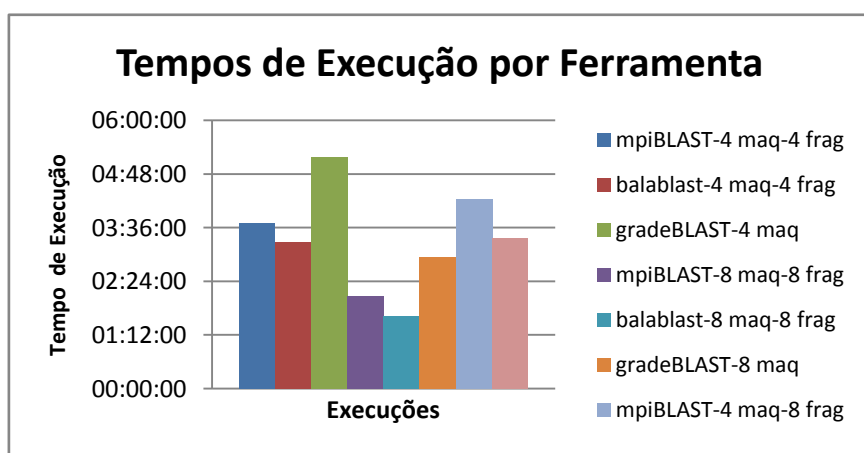


Figura 5.12: *Comparativo entre tempos de execução,, utilizando 4 ou 8 máquinas e 4 ou 8 fragmentos.*

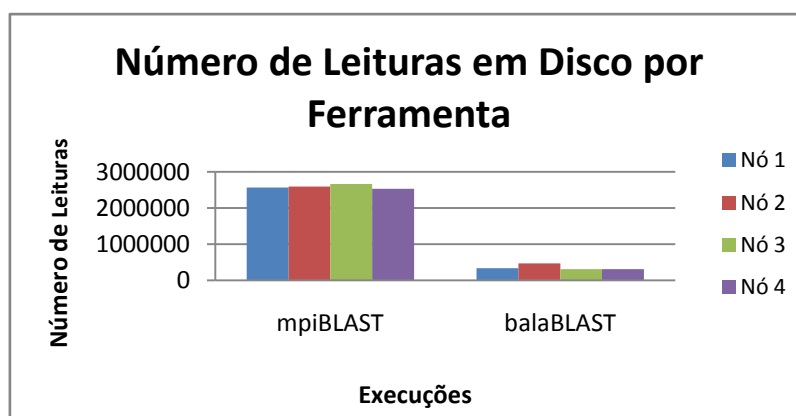


Figura 5.13: *Leituras em disco para 4 máquinas e 8 fragmentos.*



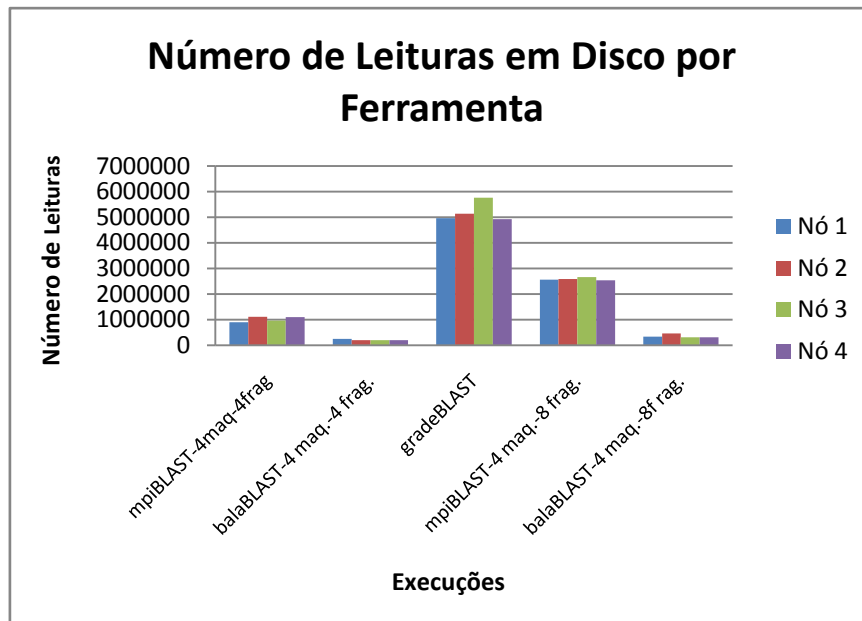


Figura 5.14: Comparativo de E/S para execução, em quatro máquinas, com 4 e 8 fragmentos.

### 5.3.4

#### Teste utilizando 8 equipamentos do Cluster da PUC-Rio para Consultas com 500 Sequências e Aproximadamente 1000 aminoácidos

Esta seção tem por objetivo analisar o desempenho das ferramentas BLAST, em avaliação, considerando um aumento do número de aminoácidos existentes em cada uma das seqüências da consulta ao BLAST constante da Seção 5.2.3, de aproximadamente 500 para aproximadamente 1000 aminoácidos.

A Figura 5.15 apresenta o desempenho das ferramentas para a consulta modificada, contendo 500 seqüências e aproximadamente 1000 aminoácidos. O balaBLAST apresentou um melhor desempenho, sendo executado em 3 horas, enquanto o mpiBLAST foi executado em 3 horas e 21 minutos, 11% superior ao balaBLAST, e o gradeBLAST em 4 horas e 10 minutos, 38% superior.

Observando-se a Figura 5.16, pode-se verificar que o balaBLAST apresentou um menor número de leituras em disco, o que explica o seu melhor desempenho em relação ao mpiBLAST e ao gradeBLAST, da mesma maneira que aconteceu nas seções anteriores.

A Figura 5.17 apresenta um gráfico comparativo de desempenho para a consulta original e a modificada. Pode-se observar que houve um incremento no tempo de execução do balaBLAST de 85%, no mpiBLAST de 63% e no gradeBLAST de 26%. No entanto, examinando os dados mais atentamente, pode-se notar que houve uma diminuição na diferença entre os tempos de execução das ferramentas em relação ao balaBLAST.

O comportamento acima descrito aponta para uma tendência de queda na eficiência do balaBLAST em relação às outras ferramentas quando o número de aminoácidos por seqüência é aumentado. Tal fato é confirmado quando tomamos as diferenças percentuais em relação ao balaBLAST para cada tipo de consulta, incluindo uma execução das ferramentas para uma consulta adicional com 500 seqüências e aproximadamente 1800 aminoácidos, conforme mostrado na Figura 5.18.

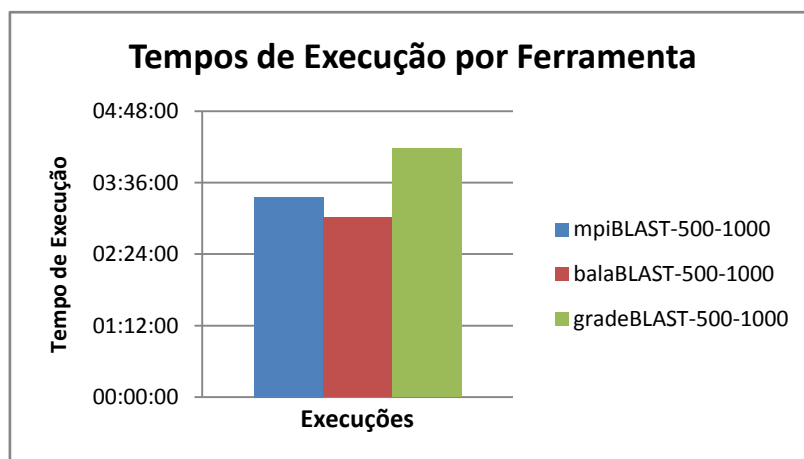


Fig 5.15: *Tempos de Execução para consulta com 500 seqüências e aproximadamente 1000 aminoácidos cada.*

A Figura 5.18 apresenta um comparativo entre as execuções considerando consultas com 500 seqüências, contendo respectivamente 500, 1000 e 1800 aminoácidos. Assim, para seqüências com 500 aminoácidos a diferença percentual entre o mpiBLAST e o balaBLAST que era de 26 % e do gradeBLAST para o balaBLAST que era de 82%, quando executando seqüências com 1800 aminoácidos, caiu para 7% considerando o mpiBLAST e 26% considerando o gradeBLAST.

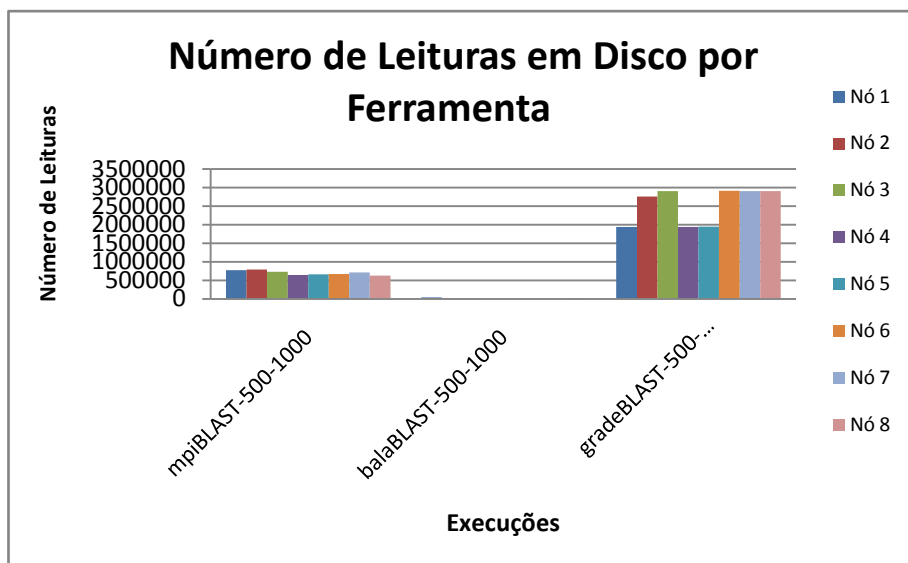


Figura 5.16: Número de leituras em disco para consulta com 500 seqüências e aproximadamente 1000 aminoácidos cada.

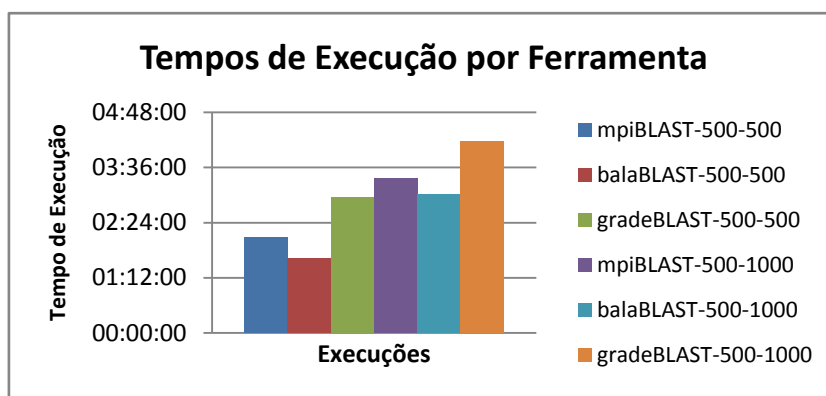


Figura 5.17: Comparativo de tempos de execução para uma consulta com 500 seqüências e aproximadamente 500 aminoácidos cada, e uma consulta com 500 seqüências e aproximadamente 1000 aminoácidos cada.

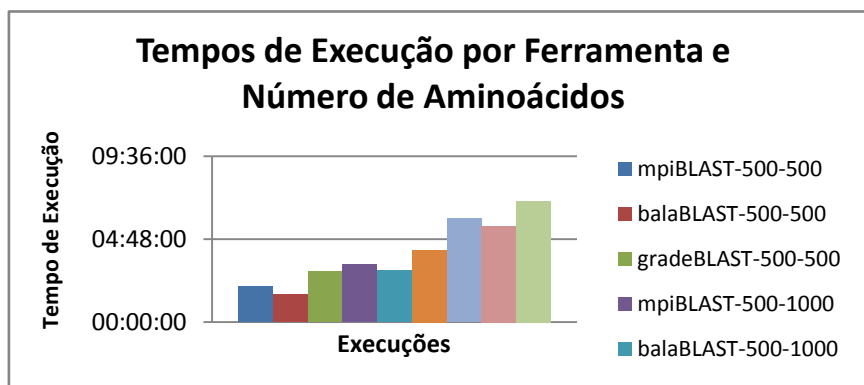


Figura 5.18: Tempos de Execução para um número crescente de aminoácidos.

### 5.3.5

#### Teste utilizando 8 equipamentos do Cluster da PUC-Rio para Consultas com 1000 Sequências e Aproximadamente 500 Aminoácidos

Nesta seção é avaliado o comportamento das ferramentas BLAST, em avaliação, considerando um aumento do número de seqüências, da consulta constante da Seção 5.2.3, de 500 para 1000 seqüências, mantido o número de aminoácidos por seqüência, ou seja, aproximadamente 500 aminoácidos por seqüência

A Figura 5.19 apresenta os tempos de execução das ferramentas para a consulta modificada. O balaBLAST mostrou ter o melhor desempenho com um tempo de execução de 3 horas e 15 minutos, enquanto o mpiBLAST foi executado em 4 horas e 5 minutos, 25% maior que o balaBLAST, e o gradeBLAST em 5 horas e 17 minutos, 62% maior que o balaBLAST, com o pior desempenho. A Figura 5,20 mostra um menor número de leituras em disco efetuadas pelo balaBLAST, o que explica seu melhor desempenho.

Uma comparação dos tempos de execução obtidos pelas ferramentas em análise considerando a consulta original e a modificada, é mostrada na Figura 5.21. Pode-se observar que os tempos de execução do balaBLAST e do mpiBLAST dobraram, enquanto o do gradeBLAST teve um aumento aproximadamente 60%.

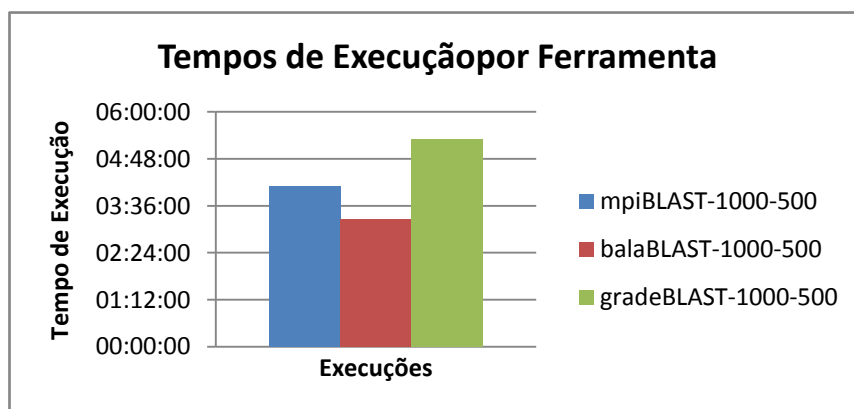


Figura. 5.19: *Tempos de Execução para Consulta com 1000 seqüências e 500 aminoácidos.*

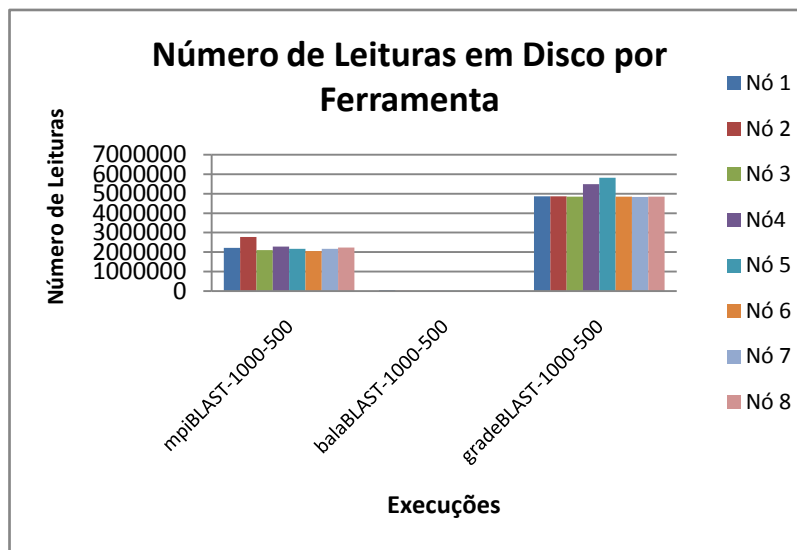


Figura 5.20: Leituras em disco efetuadas pelas ferramentas durante o processamento de uma consulta com 1000 seqüências e 500 aminoácidos.

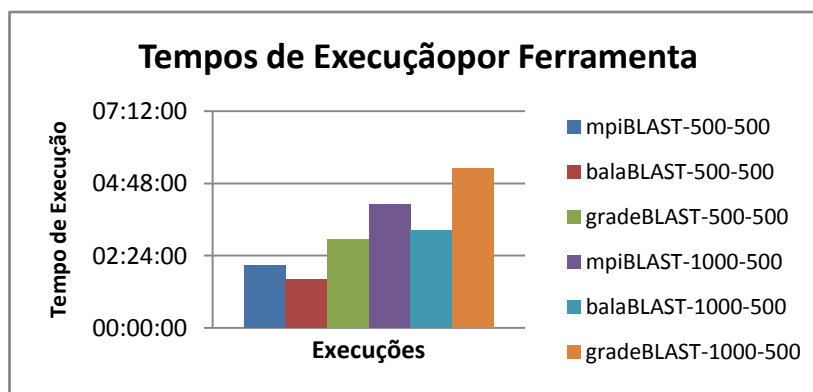


Figura 5.21: Comparativo de desempenho considerando consultas com 500 seqüências e 500 aminoácidos e 1000 seqüências e 500 aminoácidos.

### 5.3.6

#### Teste utilizando 8 Equipamentos do Cluster da PUC-Rio para Consultas com 1000 Seqüências e Aproximadamente 1000 Aminoácidos

Nas seções anteriores, foi constatado que, para consultas efetuadas utilizando 4 máquinas, tanto em ambiente de Grid, quanto de Cluster, considerando consultas contendo 500 seqüências e aproximadamente 500 aminoácidos cada, o balaBLAST obteve vantagem sobre as demais ferramentas. Da mesma forma, para execuções em 8 equipamentos da PUC-Rio, de consultas

com 500 seqüências e 1000 aminoácidos e 1000 seqüências e 500 aminoácidos, o balaBLAST também teve melhor desempenho. Em ambos os casos o melhor resultado obtido pelo balaBLAST foi motivado pela menor atividade de E/S durante a sua execução.

Nesta seção, são consideradas consultas contendo 1000 seqüências e aproximadamente 1000 aminoácidos, ou seja, o número de seqüências a examinar e o seu tamanho são duplicados em cada consulta.

A Figura 5.22 mostra os tempos de execução obtidos pelas ferramentas. O balaBLAST mostrou ser o mais rápido, com um tempo de execução de aproximadamente 5 horas e 33 minutos, contra 6 horas e 22 minutos do mpiBLAST, com um acréscimo de 14%, e 7 horas e 39 minutos do gradeBLAST, com um acréscimo de 37%. Observando a Figura 5.23, pode-se observar que o número de leituras efetuadas pelo balaBLAST foi muito menor do que as outras ferramentas, o que explica este resultado.

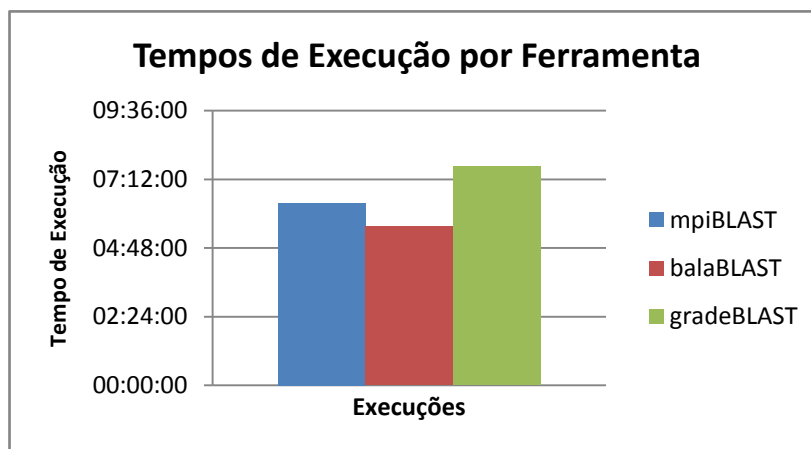


Figura 5.22: *Tempos de Execução das ferramentas considerando uma consulta com 1000 seqüências e 1000 aminoácidos.*

A Figura 5.24 mostra um comparativo de tempos de execução, considerando a consulta anterior, com 500 aminoácidos e aproximadamente 500 seqüências, e a atual com 1000 seqüências e aproximadamente 1000 aminoácidos cada. Pode-se observar que o tempo de execução das ferramentas sofreu um acréscimo médio de 200%, sendo que o maior foi o do balaBLAST com um aumento de 243%, e o menor de 159% do mpiBLAST. Tal acréscimo foi motivado em parte, pelo

aumento do número de seqüências por consulta, o que por si só justificaria a duplicação do tempo de execução. No entanto, o aumento do número de aminoácidos não deveria implicar em um acréscimo tão significativo, apesar do maior esforço do BLAST em efetuar a comparação de seqüências, principalmente no que diz respeito ao balaBLAST.

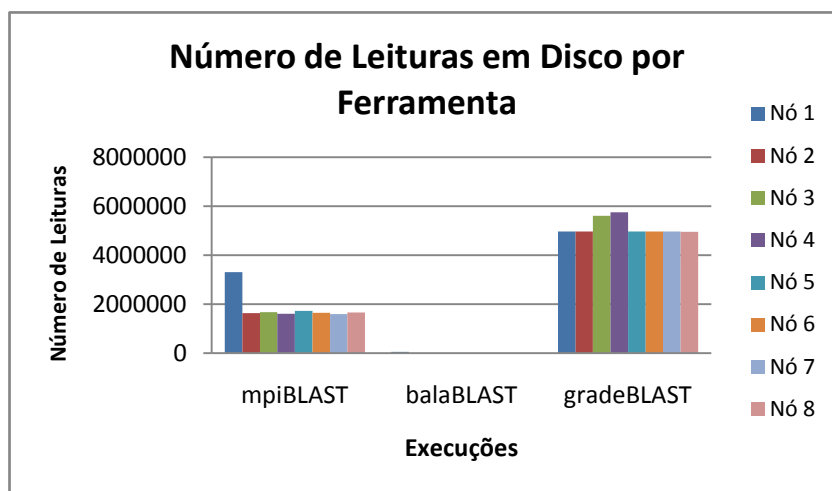


Figura 5.23: Comparativo de quantidades de leitura em disco pelas ferramentas

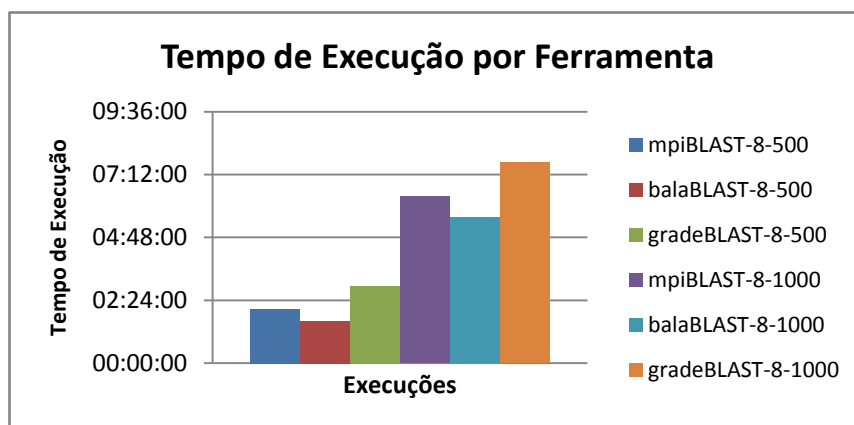


Figura 5.24: Comparativo entre tempos de execução de consultas com 500 seqüências com 500 aminoácidos, e de 1000 seqüências com 1000 aminoácidos

A Figura 5.25 apresenta o número de leituras e gravações em disco efetuadas pelo balaBLAST para 500 e 1000 seqüências. Pode-se observar que o número de leituras efetuadas não foi afetado pelo aumento do número de seqüências submetidas ao BLAST, conforme já era esperado. No entanto, é

possível notar que houve um incremento considerável, da ordem de 240%, no número gravações quando processando 1000 seqüências em relação a 500 seqüências, sendo que a diferença percentual do número de leituras e gravações para 1000 seqüências foi da ordem de 245%. Tal percentual expressa o esforço da ferramenta para elaborar o relatório consolidado com os resultados do BLAST, e explica, juntamente com o aumento do número de aminoácidos por seqüência, a elevação do tempo de execução da ferramenta.

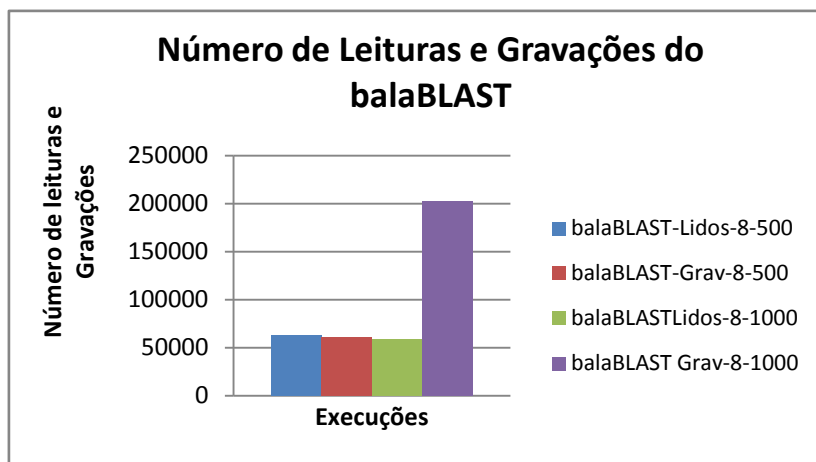


Fig 5.25: Comparativo de leituras e gravações em disco para o gradeBLAST, considerando consultas com 500 seqüências e 1000 seqüências.

No caso do mpiBLAST e do gradeBLAST, o incremento no tempo de execução foi resultado do aumento do número de aminoácidos por seqüência, tendo em vista que o número de gravações, em relação ao número de leituras permaneceu pequeno, conforme mostra a figura 5.26 e, portanto, não teve uma influência significativa no aumento do tempo de execução.

A Figura 5.27 mostra um comparativo de leituras em disco efetuadas pelas ferramentas em análise para consultas com 500 seqüências e 500 aminoácidos, e para consultas com 1000 seqüências e 1000 aminoácidos. Pode-se observar que houve um acréscimo na quantidade de leituras em disco efetuadas, de tal maneira que, no caso do mpiBLAST quase que duplicou, enquanto para o gradeBLAST mais do que duplicou.



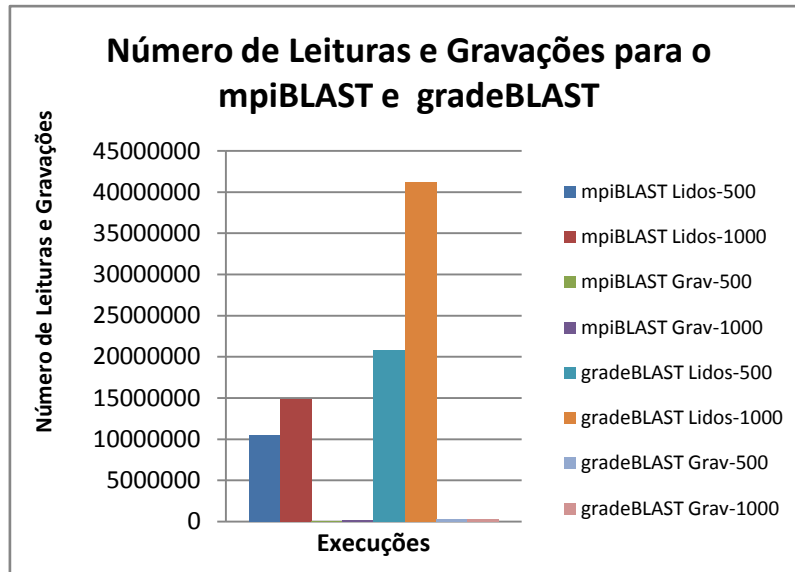


Figura 5.26: Comparativo do número de leituras e gravações em disco para o mpiBLAST e gradeBLAST, considerando consultas com 500 e 1000 seqüências.

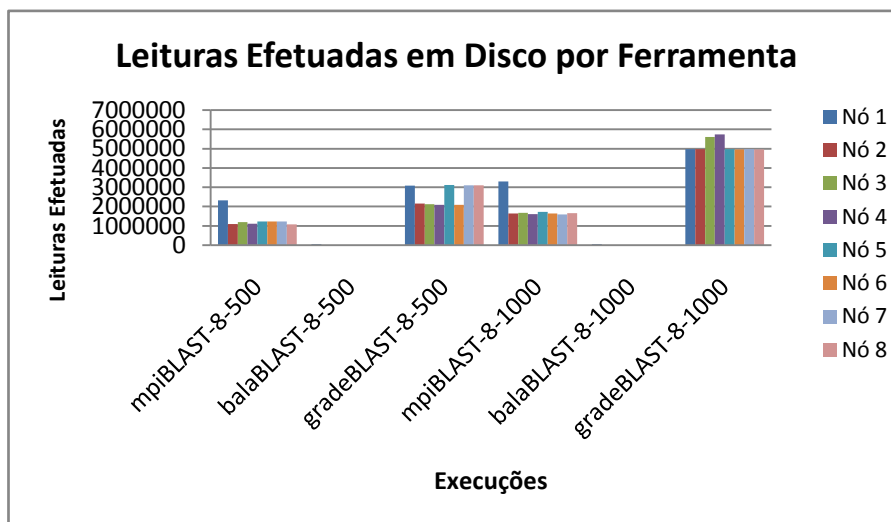


Figura 5.27: Comparativo de quantidades de leitura efetuada pelas ferramenta em cada nó, considerando consultas com 500 seqüências com aproximadamente 500 aminoácidos cada, e consultas com 1000 seqüências e aproximadamente 1000 aminoácidos cada.

### 5.3.7

#### Teste no Grid utilizando 8 Equipamentos da UFF para Consultas com 1000 Seqüências e Aproximadamente 1000 Aminoácidos

A mesma consulta da seção anterior foi efetuada utilizando um ambiente de *grid*, com duas máquinas da PUC-Rio e oito máquinas da UFF, no caso do

mpiBLAST, e uma máquina da PUC-Rio e oito máquinas da UFF, para o balaBLAST e o gradeBLAST. Infelizmente, em virtude de problemas na UFF, um dos equipamentos utilizados era do tipo 4, com um processador Intel Pentium 4 de 3 GHz e 1 GB de memória RAM.

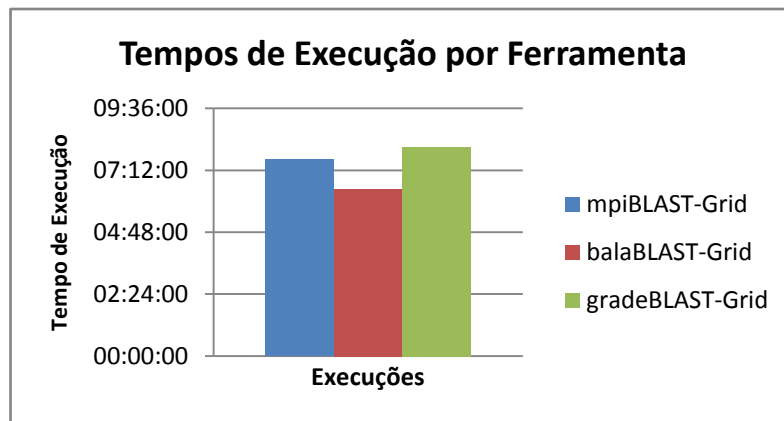


Figura 5.28: *Tempos de execução das ferramentas BLAST na UFF, considerando uma consulta com 1000 seqüências e 1000 aminoácidos.*

A Figura 5.28 mostra os tempos de execução das ferramentas BLAST considerando o ambiente de grid acima descrito. O balaBLAST foi executado em 6 horas e 28 minutos, o menor tempo de execução entre as três ferramentas, o mpiBLAST, 17% mais lento, foi executado em 7 horas e 36 minutos, e o gradeBLAST em 8 horas e 6 minutos, um tempo 25% maior do que o balaBLAST.

Vale lembrar que os equipamentos da UFF possuem uma maior capacidade de processamento e quantidade de memória do que aqueles pertencentes à PUC-Rio, o que explica um melhor resultado obtido pelo gradeBLAST em relação ao balaBLAST, que passou de um percentual de 37% de acréscimo no tempo de execução em relação ao balaBLAST para 25%. No entanto, no que se refere às estratégias fragmentadas, o que se apresentou foi uma piora no desempenho em relação à consulta anterior, como mostra a figura 5.29, sendo que houve um incremento no tempo de execução de 16,5% para o balaBLAST e de 19% para o mpiBLAST, enquanto para o gradeBLAST foi de 5,8%, inferior, portanto aos demais.

A Figura 5.30 permite a comparação das leituras efetuadas em disco durante o processamento das ferramentas em análise. O que se observa, é que o

balaBLAST necessitou efetuar um menor número de leituras em disco do que as duas outras ferramentas, cerca de 1% do total, e daí a explicação do seu melhor desempenho. O gradeBLAST, necessitou efetuar menos 57% leituras do que o mpiBLAST, incluindo a máquina “laranja”, que possui uma menor quantidade de memória do que os demais equipamentos utilizados. Este fato chama a atenção quanto à utilização de estratégias replicadas, que podem se tornar atraentes quando utilizando equipamentos de grande capacidade de processamento e memória, ainda que em um ambiente de grid, no qual a latência é considerável.

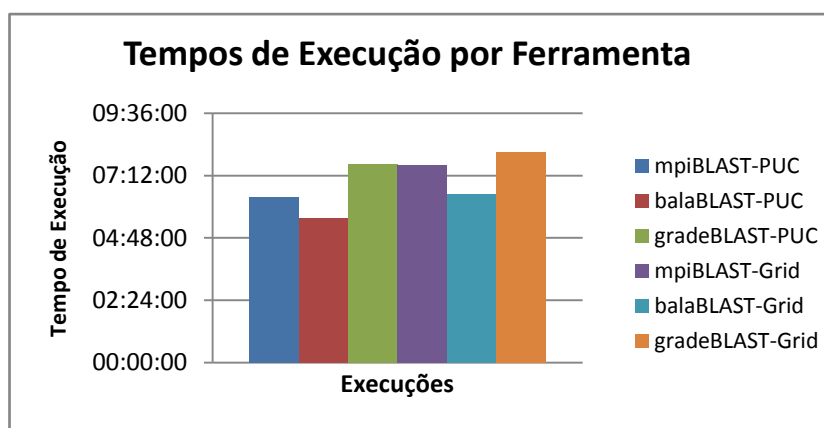


Figura 5.29: Comparação entre os *tempos de execução* considerando o teste executado na PUC-Rio com o teste executado no grid.

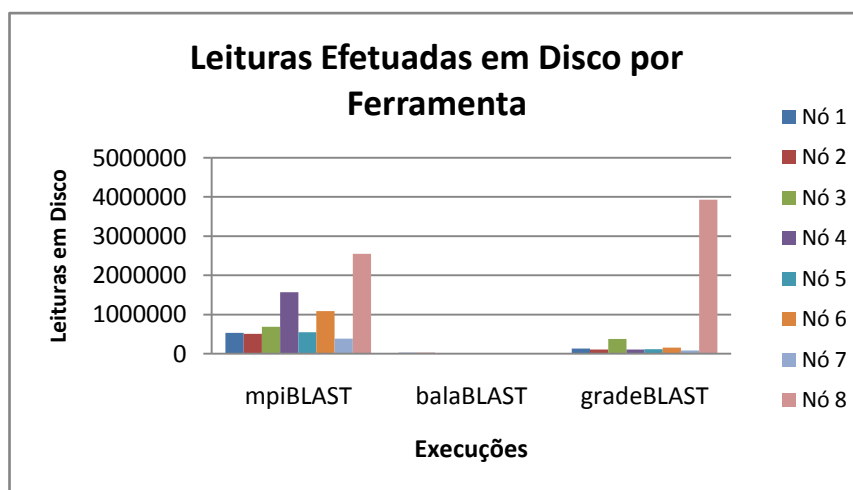


Figura 5.30: Comparativo de quantidades de leitura em disco efetuadas pelas ferramentas.

### 5.3.8

#### Teste utilizando 4 Equipamentos do Cluster da PUC-Rio para Consultas com 1000 Seqüências e 1000 Aminoácidos

A Figura 5.31, mostra os tempos de execução das ferramentas em avaliação considerando a mesma consulta da seção anterior, em quatro máquinas do tipo 1 do *cluster* da PUC-Rio. O balaBLAST mostrou ter o melhor desempenho com um tempo de execução de 11 horas e 8 minutos, enquanto o mpiBLAST processou a consulta em 11 horas e 56 minutos, 7% maior que o balaBLAST, e o gradeBLAST em 14 horas e 24 minutos, com o pior resultado.

O menor número de leituras efetuadas em disco pelo balaBLAST, apresentada na figura 5.32, pode explicar o seu melhor desempenho frente às outras ferramentas. Do total de leituras efetuadas 3% foram efetuadas pelo balaBLAST, 10% pelo mpiBLAST e 87% pelo gradeBLAST, o que justifica o pior desempenho desta última.

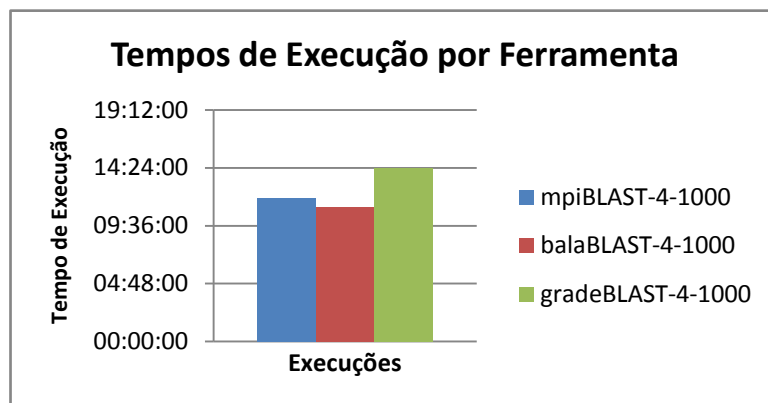


Figura 5.31: *Tempos de execução das ferramentas BLAST na PUC-Rio, considerando uma consulta com 1000 seqüências e 1000 aminoácidos, executadas em 4 equipamentos.*

A figura 5.33 apresenta um comparativo do desempenho das ferramentas, em avaliação, considerando também aquela consulta utilizada nos testes descritos na seção 5.2.1 e posteriores, contendo 500 seqüências e 500 aminoácidos. Pode-se notar que houve um acréscimo bastante acentuado no tempo de execução, em média da ordem de 217% , sendo que para o balaBLAST foi de 251%, para o

mpiBLAST de 222%, enquanto para o gradeBLAST foi de 168%, sendo esta última a que obteve menor variação.

A figura 5.34 mostra o número de leituras em disco efetuadas para as duas consultas acima. Em relação à primeira execução, o gradeBLAST foi o que apresentou um maior incremento no número de leituras efetuadas, da ordem de 100%, seguido pelo balaBLAST com 68% e pelo mpiBLAST, com apenas 2,7%. Apesar do maior crescimento em relação ao mpiBLAST, o balaBLAST foi a ferramenta que apresentou o melhor desempenho, com um menor número de leituras em disco.

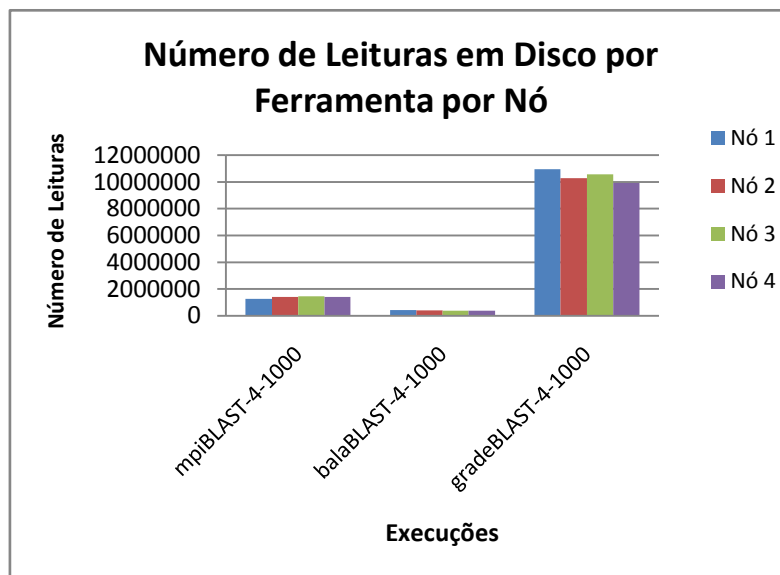


Figura 5.32: Leituras em disco efetuadas durante a execução das ferramentas para uma consulta contendo 1000 seqüências e 1000 aminoácidos por seqüência.

Observando-se as estatísticas envolvidas, pode-se concluir, que à excessão do gradeBLAST, o incremento no tempo de execução das ferramentas foi motivado principalmente pelo esforço computacional em processar as consultas, e não propriamente pelo número de acessos a disco envolvidos nesse processamento.

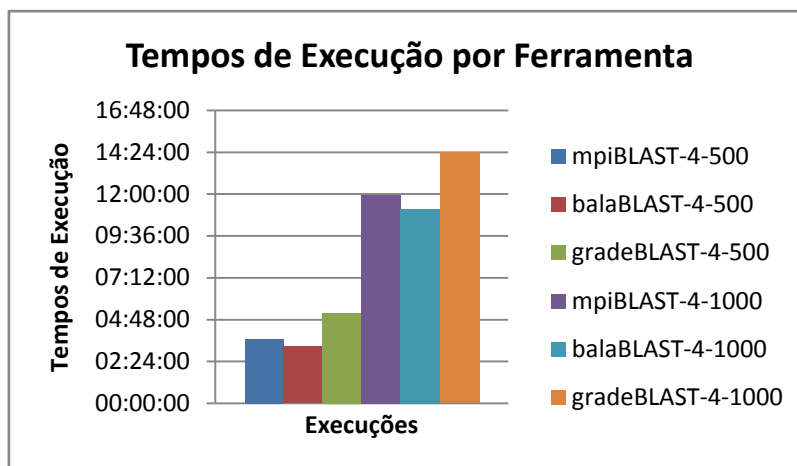


Figura 5.33: Comparativo dos tempos de execução das ferramentas considerando uma consulta com 500 seqüências com 500 aminoácidos e uma consulta com 1000 seqüências e 1000 aminoácidos por seqüência.

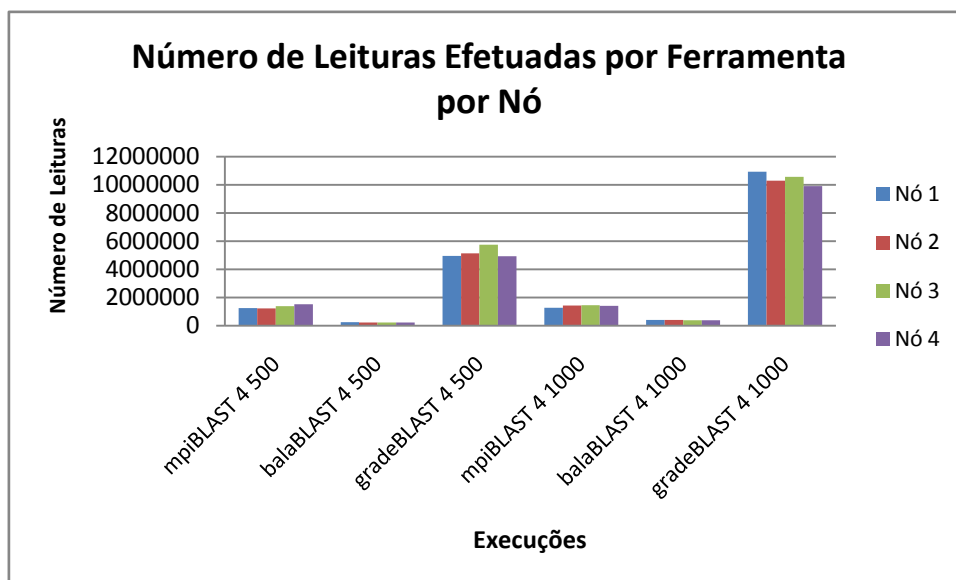


Figura 5.34: Comparativo de leituras em disco efetuadas pelas ferramentas BLAST considerando uma consulta com 500 seqüências e 500 aminoácidos, e uma consulta com 1000 seqüências e 1000 aminoácidos.

### 5.3.9

#### Teste utilizando 4 Equipamentos do Cluster da PUC-Rio para Consultas contra a Base Swissprot

Embora as avaliações que foram efetuadas durante o presente trabalho tenham se pautado em uma base de dados biológica de grande porte, é interessante

verificar também o comportamento das ferramentas em análise quando efetuando o processamento em bases menores, como é o caso da base de proteínas Swissprot.

A Figura 5.35 apresenta os tempos de execução de cada uma das ferramentas para uma consulta retirada da base Swissprot, contendo 500 seqüências com aproximadamente 500 aminoácidos cada, utilizando 4 máquinas do tipo 1, e 4 fragmentos. Pode-se observar que o melhor desempenho foi obtido não pelo balaBLAST, como nos testes anteriores, mas sim pelo mpiBLAST, com um tempo de execução de aproximadamente 6 minutos e 31 segundos contra 7 minutos e 21 segundos do balaBLAST e 11 minutos e 52 segundos do gradeBLAST.

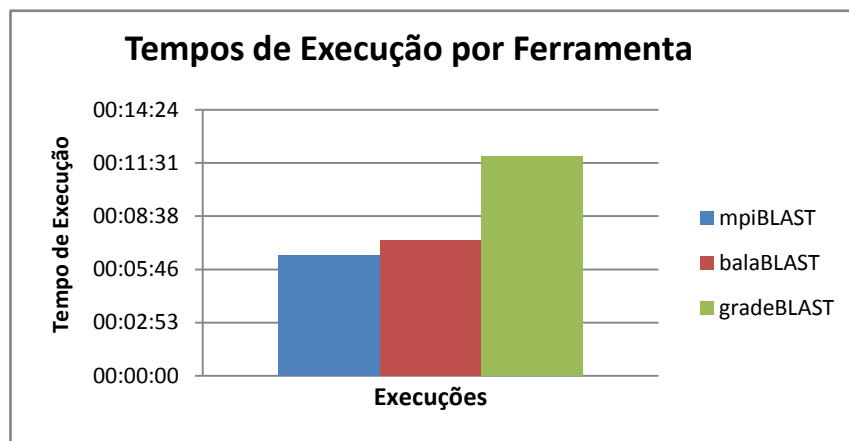


Figura 5.35: *Tempos de Execução das ferramentas para consulta com 500 seqüências e aproximadamente 500 aminoácidos cada.*

Quando aumentado o número de seqüências de consultas para 1000, com aproximadamente 1000 aminoácidos cada uma, os tempos de execução apresentaram o comportamento mostrado na Figura 5.36, em que o balaBLAST teve o pior resultado entre as ferramentas analisadas, com um tempo de execução de aproximadamente 28 minutos, enquanto o do mpiBLAST foi de 12 minutos, e o do gradeBLAST foi de 20 minutos.

A Figura 5.37 mostra que a quantidade de leituras efetuadas pode ser considerada desprezível em relação ao número de gravações quando tratando bases menores, como a Swissprot, podendo com isso serem desconsideradas para

análise. Como o maior esforço encontra-se na elaboração do relatório com os resultados do BLAST, então apenas o número de gravações deve ser considerado.

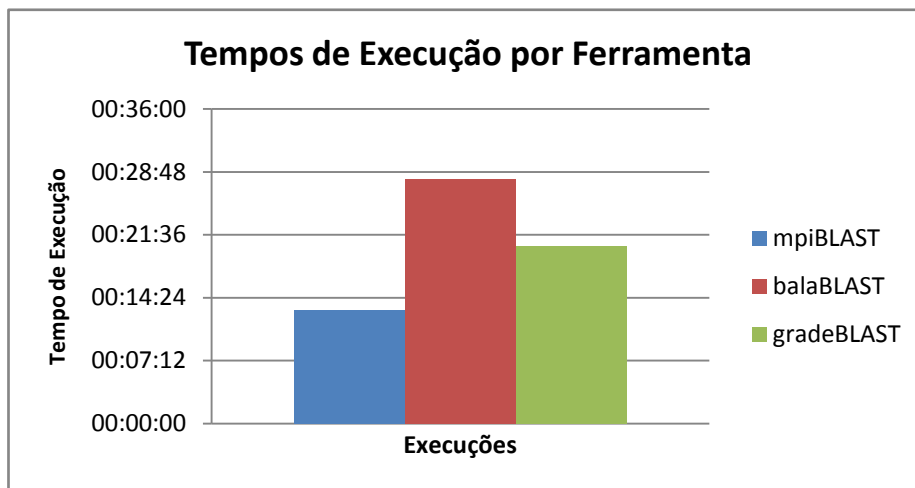


Figura 5.36: Tempos de Execução, considerando consultas com 1000 seqüências com aproximadamente 1000 aminoácido cada.

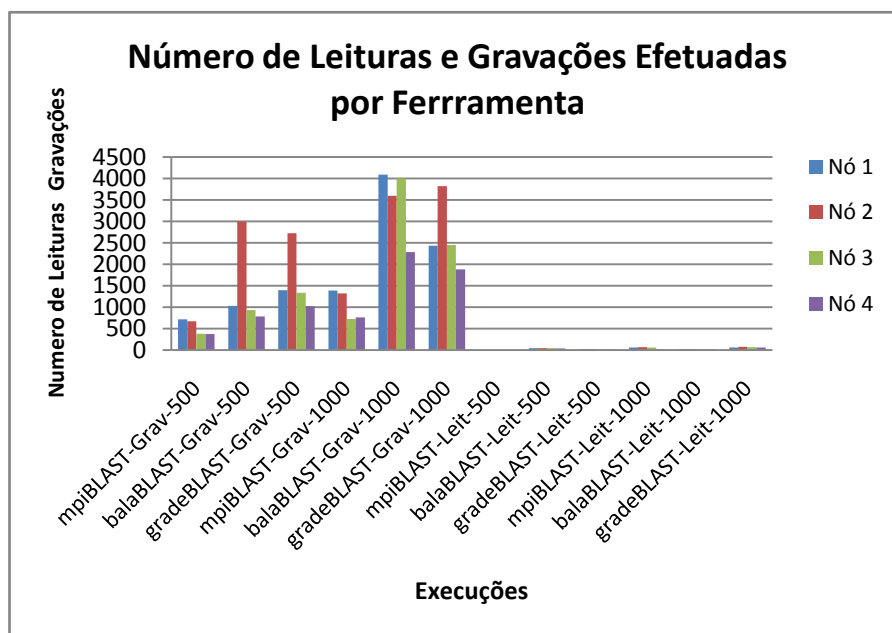


Figura 5.37: Comparativo entre as quantidades de gravações e leituras efetuadas na execução de uma consulta de 1000 seqüências e aproximadamente 1000 aminoácidos.

As Figuras 5.38 e 5.39 mostram as atividades de E/S, relativas às gravações efetuadas pelas ferramentas, em cada um dos nós trabalhadores, durante o



processamento das consultas contendo 500 seqüências, e 1000 seqüências. No processamento da consulta contendo 500 seqüências, o número de gravações efetuadas durante a execução do balaBLAST, foi maior que a do mpiBLAST. Na consulta envolvendo 1000 seqüências, o balaBLAST efetuou o maior número de gravações em disco entre as ferramentas em avaliação.

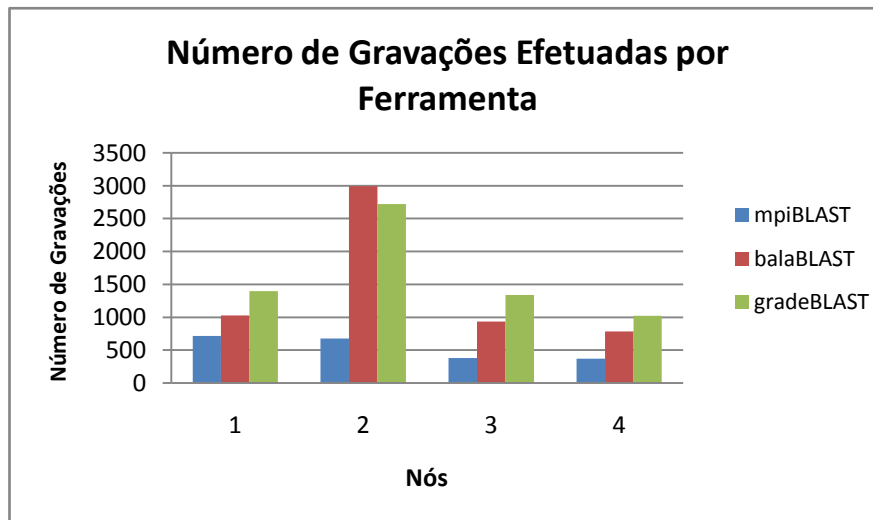


Figura 5.38: *Quantidade de gravações efetuadas em cada um dos nós trabalhadores, para consulta com 500 aminoácidos e aproximadamente 500 seqüências, incluindo*

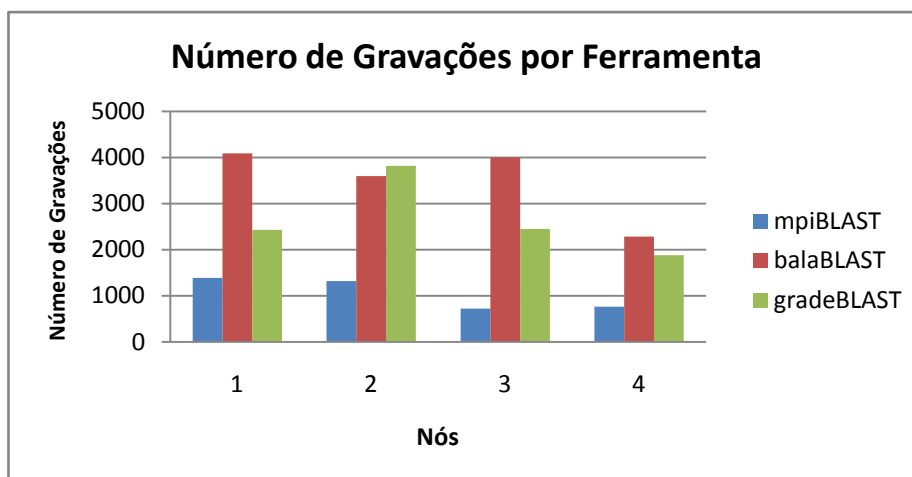


Figura 5.39: *Quantidade de gravações efetuadas, em cada um dos nós trabalhadores, para uma consulta de 1000 seqüências e aproximadamente 1000 aminoácidos.*

As Figuras 5.40 e 5.41 incluem as gravações efetuadas no nó mestre durante a execução das ferramentas. Pode-se observar que o número de gravações efetuadas no nó mestre quando utilizando balaBLAST foi muito maior que nas

demais ferramentas. Este crescimento foi bastante acentuado quando o número de seqüências e o número de aminoácidos por seqüências passou de 500 para 1000, o que não aconteceu com o mpiBLAST e o gradeBLAST, cuja variação não chegou a ser duplicada.

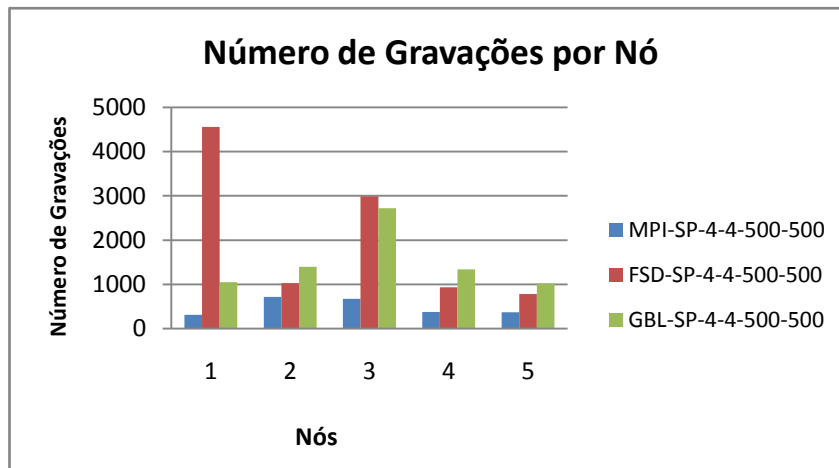


Figura 5.40: Quantidade de gravações efetuadas, em cada um dos nós, incluindo o nó mestre,, para uma consulta de 500 seqüências e aproximadamente 500 aminoácidos.

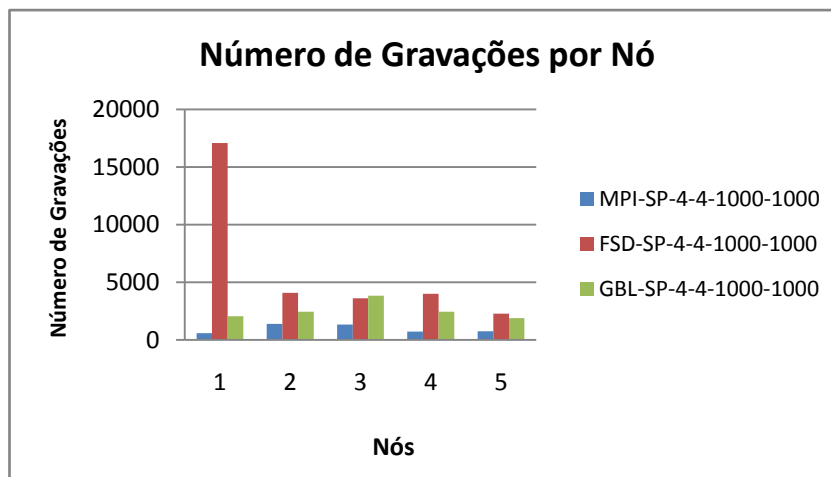


Figura 5.41: Quantidade de gravações efetuadas, em cada um dos nós, incluindo o nó mestre,, para uma consulta de 1000 seqüências e aproximadamente 1000 aminoácidos.

A Figura 5.42 mostra o número de gravações efetuadas considerando o conjunto dos nós utilizados durante a execução das ferramentas. A variação do número de gravações efetuadas durante a execução do balaBLAST para consultas de 1000 seqüências e aproximadamente 1000 aminoácidos, em relação àquelas

com 500 seqüências e aproximadamente 500 aminoácidos, foi de 200% , enquanto para o mpiBLAST foi de 94% e para o gradeBLAST de 67%.

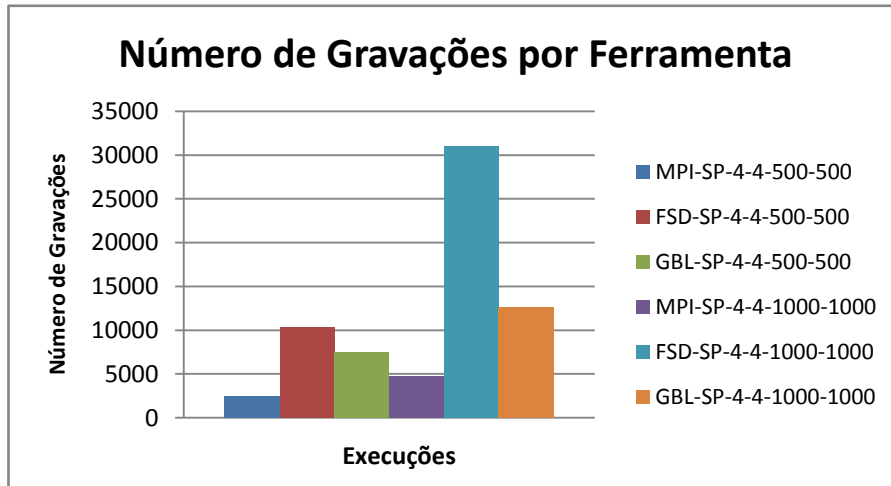


Figura 5.42: Comparativo do número total de gravações efetuadas, incluindo o nó mestre,, para uma consulta de 500 seqüências e aproximadamente 500 aminoácidos,e uma consulta de 1000 seqüências e aproximadamente 1000 aminoácidos.

Considerando o tamanho da base Swissprot,em relação à base NR, pode-se observar que o número de gravações efetuadas no presente teste reflete basicamente o esforço das ferramentas na elaboração do relatório final de saída contendo os resultados do processamento das ferramentas. e que pode influenciar no desempenho final de uma ferramenta BLAST em paralelo.

Uma das preocupações citadas em [28] diz respeito à elaboração do relatório de saída, o qual consolida os resultados parciais obtidos, em uma estratégia fragmentada, sobre cada fragmento da base de dados biológica em cada um dos nós, durante a execução da ferramenta BLAST em paralelo.

No caso do balaBLAST essa consolidação é efetuada apenas pelo nó mestre, enquanto no mpiBLAST, tal tarefa é dividida entre o nó mestre e os nós trabalhadores. No mpiBLAST, após o processo de pesquisa sobre um fragmento, os nós trabalhadores convertem seus resultados intermediários no formato final de saída, e enviam os metadados de saída para o nó mestre,que por sua vez, efetua o cálculo dos deslocamentos de saída para cada registro no arquivo de saída global e envia a esta informação para os nós trabalhadores, que de posse das informações

de deslocamento são capazes de gravar registros de saída em paralelo para o sistema de arquivos através a interface MPI-IO. Tal procedimento acelera o desempenho da ferramenta mpiBLAST em relação ao balaBLAST, o que pode explicar os resultados obtidos pelo mpiBLAST nos testes.

No caso do gradeBLAST, como a base de comparação não são os fragmentos, mas sim, a própria base biológica replicada, não há necessidade de uma consolidação de resultados parciais obtidos para cada seqüência de consulta, mas, simplesmente, uma concatenação dos resultados obtidos em cada um dos nós trabalhadores, que são enviados ao mestre, o qual efetua tal tarefa.

## 5.4

### Conclusão

A avaliação de desempenho das ferramentas BLAST em paralelo analisadas, ou benchmark, demonstrou uma performance bastante semelhante do mpiBLAST e do balaBLAST. No entanto, em virtude da estratégia de E/S adotada, o balaBLAST mostrou ter o melhor desempenho. A comparação levou em consideração uma base de dados com um volume de dados considerável, e que é bastante utilizada entre os biólogos, que é a base NR.

O mau resultado obtido pelo balaBLAST quando executado considerando uma base Swissprot, de pequeno porte, não contradiz a afirmação acima, mesmo porque só existe sentido em se executar uma estratégia em paralelo para grandes bases de dados biológicas.

O resultado obtido na estratégia replicada para uma grande base de dados, no entanto, não foi tão bom quanto as duas outras. Tal fato é explicado ao se ponderar o tamanho da base analisada, o tamanho da memória RAM de cada equipamento e as poucas máquinas efetivamente disponíveis, o que, em conjunto, obriga o sistema a efetuar paginações de memória, e em consequência, atrasa o processamento das seqüências. A estratégia fragmentada, ao contrário, leva alguma vantagem por necessitar de menos quantidade de memória RAM, em

virtude de a base encontrar-se dividida pelos nós, e com isso necessitar de pouca, ou nenhuma, paginação de memória.

No entanto, para bases menores que a NR, dependendo do tamanho da base e do número de consultas efetuadas, além da complexidade em se consolidar resultados em uma estratégia fragmentada [18], a estratégia replicada, talvez seja bem interessante, principalmente quando tratando de simplicidade, embora perca em termos de tempo de execução para o mpíBLAST.

Assim, para bases de dados do porte da NR, ou maiores, a adoção de uma estratégia fragmentada é a melhor opção. A análise de fragmentos da base de dados alocados em cada um dos equipamentos é mais eficiente do que avaliar a base inteira, principalmente considerando o número de acessos a disco necessários durante o processamento, como ocorre em estratégias de replicação da base. Entre as ferramentas que utilizam a estratégia fragmentada, o balaBLAST foi aquela que apresentou um melhor desempenho nos testes efetuados.