

4.

Balanceamento de Carga

Pode-se entender por balanceamento de carga uma política a ser adotada para minimizar tanto a ociosidade de utilização de alguns equipamentos quanto a super utilização de outros, em um ambiente distribuído [31]. O objetivo que se tem, ao efetuar esse procedimento, é prover uma melhor distribuição na utilização de recursos em aplicações paralelas, e, em consequência, possibilitar uma maior eficiência em sua utilização [31].

Um dos objetivos do presente trabalho, e que constitui o teor do presente capítulo, é a avaliação da ferramenta balaBLAST quanto ao balanceamento de carga. Assim, são aqui apresentados alguns testes efetuados com a finalidade de verificar se a ferramenta balaBLAST realmente efetua uma distribuição de carga pelos diferentes nós em um ambiente distribuído. Para tal, são consideradas diferentes situações em ambientes de *cluster* e de *grid*, que permitem que se faça esta verificação.

4.1

Balanceamento de Carga em Ferramentas BLAST

Ambientes de clusters e grids são utilizados para efetuar análise de similaridades utilizando-se ferramentas BLAST em paralelo [4, 5], no entanto, a simples utilização do paralelismo não traz, por si só, a eficiência na execução dessas ferramentas, principalmente considerando que fazem uso intensivo de E/S. O próprio conteúdo das seqüências de consulta, além do tamanho, conduz ao desequilíbrio na utilização dos recursos (desvio de similaridade), o que de certa forma contraria o fato que aplicações BLAST sejam facilmente paralelizáveis.

Da mesma forma, diferentes equipamentos com diferentes recursos podem, também, traduzir diferenças de desempenho no trato com os dados. Dessa maneira, equipamentos que possuem discos mais rápidos dotados de maior capacidade de armazenamento, com maiores quantidades de memória RAM, associados a um processador de última geração, terão um desempenho melhor que algum outro que possua recursos mais modestos (desvio de heterogeneidade).

Além disso, a disputa por utilização de recursos em um ambiente distribuído, principalmente em um ambiente multiusuário, pode acarretar em não utilização da capacidade plena dos equipamentos pertencentes àquele ambiente, o que pode acarretar em atrasos no processamento (desvio de concorrência).

Por outro lado, a utilização de ambientes distribuídos, ainda que homogêneos (máquinas com mesma localização, porte e sistema operacional), por si só, não se traduz em eficiência na utilização dos recursos computacionais durante a execução da ferramenta BLAST. Um dos motivos para que isso aconteça, está relacionado ao fato de que para um conjunto de seqüências de consulta, de mesmo tamanho, submetidas ao BLAST, considerando uma mesma base de dados biológica, a quantidade de *hits* obtida individualmente, durante o processamento, irá depender do conteúdo das bases nucleotídicas ou aminoácidos existentes em cada uma das seqüências.

Como o tempo de processamento da ferramenta BLAST depende também do conteúdo das base nucleotídicas e dos aminoácidos, existentes nas seqüências de consulta, pode-se afirmar que tanto em ambientes homogêneos, quanto heterogêneos, a ordem com que as seqüências de consulta são submetidas pode influenciar a eficiência da execução.

4.2

Balanceamento de Carga Sob Demanda

Entende-se por carga, em execuções da ferramenta BLAST, o número total de seqüências que uma máquina gerente envia às máquinas trabalhadoras, durante todo o processamento [19]. A cada tarefa enviada às estações de trabalho existe,

portanto, um incremento da carga. Como tarefa entende-se um conjunto de seqüências submetidas de uma só vez ao BLAST.

Um dos objetivos estabelecidos para o presente trabalho é a verificação de uma das propostas de balanceamento propostas por Sousa [19], que tem por base a distribuição de tarefas denominada Fragmentada Sob Demanda (Seção 3.3), utilizada pelo balaBLAST, a partir de uma série de testes efetuados em ambientes de *cluster* e *grid*.

No entanto, na presente dissertação não existe interesse em medir a eficiência da ferramenta em si, mas verificar se ela está efetuando uma distribuição de carga eficaz pelos nós pertencentes ao ambiente distribuído, mesmo porque a aferição de ganhos de desempenho já foi efetuada em [19].

No mesmo trabalho, Sousa propõe outra abordagem, a qual denominou de Fragmentada Corretiva (Seção 3.3). Nela, a partir da verificação de que uma máquina está muito lenta ou apresentando algum problema, o nó Mestre transfere um conjunto de seqüências, que seriam processadas naquela máquina para outra, em condições de efetuar aquele processamento. O resultado é a obtenção de uma maior eficiência, sem prejuízo do resultado desejado.

Como já foi mencionado, na proposta Fragmentada Sob Demanda, objeto de verificação neste trabalho, uma base de dados biológica, por exemplo, uma base não redundante de proteínas NR, é dividida em um determinado número de partes, ou fragmentos, que são distribuídos entre os diversos nós de um *cluster*. Todos os fragmentos são copiados para todos os nós. Em cada um dos nós existe pelo menos um fragmento que serve de base ao processamento de tarefas destinadas àquele nó, que é denominado Fragmento Primário. Os demais fragmentos, chamados Fragmentos Secundários, podem ser utilizados em casos de falha no processamento, ou quando for possível acelerar o processamento de uma tarefa, utilizando algum outro nó que esteja com pouca carga.

Fixado um número de seqüências que devem ser submetidas ao BLAST de uma só vez, as tarefas são então atribuídas a cada nó trabalhador. Inicialmente cada um dos nós recebe as mesmas seqüências que os demais. A partir daí as tarefas são submetidas para os nós à medida que as tarefas anteriores sejam completadas. Caso algum nó não possua alguma tarefa atribuída, pode assumir

alguma que tenha sido inicialmente designada a algum outro, de modo a tornar mais eficiente o processamento.

4.3

Testes com a Ferramenta balaBLAST utilizando Estratégia Sob Demanda

Inicialmente, para se efetuar a verificação do balanceamento de carga da ferramenta balaBLAST, conforme sugerido na Seção 2.2, foi necessário escolher alguma métrica que fosse capaz de indicar, de forma significativa, se o balanceamento de carga está, ou não, sendo efetuado, e que atendesse as características citadas na Seção 2.1.

Um dos fatores a ponderar para a escolha de uma métrica é o tipo de ambiente no qual uma ferramenta é normalmente executada. O balaBLAST pode ser executado tanto em um ambiente de *cluster* quanto de *grid*. Assim, o potencial de carga a distribuir vai depender bastante, mas não somente, da capacidade de processamento de cada uma das máquinas pertencentes a esse ambiente. Dessa maneira, a tendência é que em máquinas com maior capacidade de processamento sejam processadas mais seqüências de consulta do que em outras com menor capacidade, ou seja, o balanceamento de carga é efetuado partindo-se da premissa que a ferramenta deve distribuir as tarefas de acordo com a capacidade de cada equipamento. Portanto, pôde-se apontar como métrica adequada para efetuar a verificação do balanceamento, o “Número de Tarefas” executadas em cada nó pertencente ao ambiente distribuído.

A manutenção de um mesmo nível quantitativo de tarefas por nó, quando da execução do balaBLAST em um ambiente homogêneo (equipamentos idênticos utilizando o mesmo sistema operacional), não evidencia que o balanceamento esteja sendo executado, mesmo porque a carga de execução é dependente do conteúdo das seqüências de consulta. Por outro lado, a execução da ferramenta em um ambiente heterogêneo torna mais evidente a distribuição de carga, principalmente quando o número de tarefas processadas em equipamentos de menor capacidade é bem menor do que em outros de maior capacidade.

A utilização da métrica “Numero de E/S”, ou seja, a análise do número de leituras e gravações em disco efetuadas durante a execução do balaBLAST, pode ajudar a esclarecer o comportamento da execução do BLAST em cada um dos nós de um ambiente distribuído, indicando o porquê do bom, ou mal, desempenho em cada um dos equipamentos. Assim, por exemplo, em um equipamento que possui pouca memória RAM, o número de leituras e gravações efetuadas em disco deverá ser bem maior do que em equipamentos com maior capacidade de memória, o que acarreta um maior tempo de processamento das seqüências de consulta a ele distribuídas. Assim, a métrica “Numero de E/S” foi também adotada.

Para medição do número de E/S, durante a execução do BLAST, foi utilizada uma ferramenta, capaz de recuperar tal informação a partir de um arquivo virtual situado no diretório “/proc” do Linux.

O Anexo 4 contém uma tabela com os testes que foram efetuados na presente avaliação, e que consideram execuções nos ambientes de *cluster* e de *grid*. As execuções foram planejadas de modo a explorar a heterogeneidade dos equipamentos assim como de ambientes de software. Dessa forma, foram utilizados equipamentos mais modernos, baseados na tecnologia *multi-core*, assim como equipamentos mais antigos, baseados no processador Pentium 4. Alguns testes em *grid* foram também realizados, para verificar a funcionalidade do balanceamento nesse ambiente. Os equipamentos utilizados nos testes de verificação, situados na Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) e na Universidade federal Fluminense (UFF), constam da tabela 4.1.

Considerando-se que a estratégia utilizada pelo balaBLAST, para implementar a paralelização do BLAST, é a fragmentação da base de dados biológica, e que possivelmente a distribuição das seqüências pertencentes a cada fragmento pode, de alguma forma, ter influência no tempo de execução da ferramenta, adotamos 3 critérios de particionamento da base de dados nos testes efetuados na presente dissertação. A motivação para tal procedimento é verificar o comportamento da estratégia de balanceamento de carga face a cada um dos seguintes tipos de particionamento utilizados:

- a) Particionamento normal:

A base de dados é particionada de acordo com a ordem dos registros na base de dados;

b) Particionamento randômico:

A base de dados é particionada de forma aleatória, de modo que o número de seqüências em cada fragmento seja aleatório, assim como o número de bases nucleotídicas ou aminoácidos em cada seqüência.

c) Particionamento balanceado:

A base de dados é particionada de modo que os fragmentos possuam aproximadamente o mesmo número de seqüências e as seqüências, em cada um dos fragmentos, tenham aproximadamente o mesmo tamanho.

Instituição	Equipamentos
PUC-Rio Tipo 1	12 CPUs Intel Core 2 Duo 2.16 GHz RAM 1 GB, Fedora Linux 8
PUC-Rio Tipo 2	20 CPUs Intel Pentium IV 1.70 GHz e 256 Mb, Red Hat Linux 9
UFF Tipo 3	13 CPUs Intel Core Quad 2.4 GHz , RAM 3 GB, HD 220 GB, Ubuntu Linux 8.04.1
UFF Tipo 4	3 CPUs Intel Pentium IV 3.2 GHz, RAM 1 GB, HD 71 GB, Fedora 4.

Tabela 4.1: *Recursos computacionais utilizados.*

À exceção dos testes realizados em ambientes de *grid*, tipicamente não controlado, todos os demais testes de balanceamento de carga foram efetuados no cluster da PUC-Rio em modo exclusivo.

As bases de dados biológicas consideradas na elaboração dos testes de balanceamento de carga foram: a base de proteínas não redundante NR, que possuía aproximadamente 4 GB e 7 milhões de registros em Novembro de 2008, e a base de nucleotídeos NT com aproximadamente 20 GB, e aproximadamente 5 milhões de registros em Julho de 2007.

4.3.1

Testes utilizando Somente Equipamentos do Cluster da PUC-Rio – Equipamentos Heterogêneos

A Listagem 4.1 mostra alguns dados estatísticos obtidos a partir da execução do BLAST para a análise de similaridade de um conjunto de 1000 seqüências com 500 aminoácidos cada, contra uma base não redundante de proteínas NR, utilizando 12 máquinas, sendo 8 máquinas com maior capacidade de processamento (tipo 1) e 4 com menor capacidade (tipo 2), no *cluster* da PUC-Rio.

A coluna “Tarefas” discrimina o número de tarefas que foram processadas por cada um dos equipamentos, e constitui a métrica adotada para a avaliação do balanceamento de carga. A numeração de 1 a 8 corresponde ao conjunto de máquinas que possuem maior capacidade de processamento que as quatro últimas, numeradas de 9 a 12.

A coluna “Conclusão” discrimina o tempo de conclusão das tarefas atribuídas a cada equipamento, em segundos. Abaixo dessa coluna é mostrado o tempo total de execução da ferramenta. Pode-se observar que o tipo de particionamento da base de dados utilizado não influenciou diretamente o tempo total de execução da ferramenta, que foram muito próximos, sendo que para o particionamento balanceado e randômico foi de 3 horas e 10 minutos, enquanto para o particionamento normal foi de 3 horas e 12 minutos, ou seja, um percentual superior apenas 1% em relação ao particionamento randômico.

SQ_RP_SP_1000_500 - Normal:RP_NR_12_12_FSD_SP_1000_500

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	11034,91	11544,75	11006,63	538,09	54
2	11045,50	11544,32	11013,65	530,64	54
3	10909,92	11544,35	10881,83	662,43	53
4	11040,25	11543,92	11010,12	533,77	54
5	10903,15	11544,74	10869,80	674,91	53
6	11041,66	11544,25	11013,53	530,72	54
7	10900,72	11544,16	10867,81	676,31	53
8	11030,80	11544,09	11001,55	542,52	54
9	10914,73	11545,17	10900,45	643,84	18
10	11352,56	11545,00	11338,98	205,54	19
11	11226,80	11545,01	11213,12	331,74	19
12	11253,91	11545,04	11238,88	305,81	19

FSD 11574,16 11351,96 192,43 29,52
 SQ_RP_SP_1000_500 - Balanceado: RP_NR_12_12_FSD_SP_1000_500

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	10918,68	11409,60	10889,42	520,13	53
2	11053,89	11409,18	11022,80	386,33	54
3	10920,89	11409,21	10891,60	517,60	53
4	11048,84	11408,79	11016,07	392,68	54
5	11063,41	11409,60	11029,92	379,66	54
6	11046,68	11408,89	11015,66	393,19	54
7	11079,89	11409,02	11044,29	364,73	54
8	11052,25	11408,95	11015,82	393,09	54
9	10929,66	11410,03	10914,29	495,01	18
10	10996,72	11409,87	10981,80	427,49	18
11	11216,76	11409,85	11203,49	206,08	19
12	11206,64	11409,91	11191,86	217,66	19

FSD 11441,33 11216,12 193,09 31,83
 SQ_RP_SP_1000_500 - Randomico: RP_NR_12_12_FSD_SP_1000_500

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	10906,88	11412,57	10878,47	534,06	53
2	10908,95	11413,47	10878,60	534,82	53
3	11054,58	11413,59	11024,04	389,51	54
4	11030,61	11413,13	11000,30	412,78	54
5	11042,29	11412,93	11012,39	400,51	54
6	11026,15	11412,84	10994,54	418,28	54
7	10910,24	11412,38	10879,30	533,04	53
8	11062,91	11413,28	11035,22	378,02	54
9	10906,95	11411,84	10889,39	521,69	18
10	11181,12	11411,89	11167,00	244,68	19
11	11220,16	11412,93	11205,82	206,89	19
12	11171,02	11413,69	11154,89	257,97	19

FSD 11440,82 11219,05 192,74 28,77

Listagem 4.1: Execução da Estratégia Sob Demanda para uma base NR em 12 máquinas, sendo 8 do tipo 1 e 4 do tipo 2, para consulta com 1000 seqüências e 500 bases cada.

As Figuras 4.1 e 4.2, baseadas na Listagem 4.1, apresentam visualmente os resultados obtidos. A Figura 4.1 mostra que o número de tarefas designadas para os nós de maior capacidade de processamento foi algo em torno de 50 tarefas por nó, enquanto para os de menor capacidade foi em torno de 18 tarefas, ou seja, menos de 50 %. A Figura 4.2 mostra que os tempos de execução das máquinas de menor capacidade foram maiores do que as demais, à exceção do nó de número 9, que processou um número menor de tarefas.

Apesar da diferença de desempenho dos equipamentos envolvidos no processamento, evidenciado nas Figuras 4.1 e 4.2, o balaBLAST efetuou o balanceamento de carga, e os nós com a mesma capacidade de processamento

computaram um número semelhante de tarefas. Em média, os nós mais rápidos processaram cada uma das tarefas em 3,42 minutos, enquanto os nós mais lentos o fizeram em 9,88 minutos. O desvio padrão calculado em 3 minutos, tomando por base o tempo médio de execução de apenas uma tarefa por máquina, indica um desempenho ruim na execução das tarefas em função das máquinas de menor capacidade computacional utilizadas no teste, mas que foi minimizado em função do balanceamento de carga efetuado.

A Listagem 4.2 mostra os resultados estatísticos obtidos considerando a base de dados de nucleotídeos NT, e as mesmas máquinas utilizadas no teste anterior, para um conjunto de 500 seqüências de consulta com aproximadamente 500 bases nucleotídicas cada.

Observando a coluna “Tarefas”, percebe-se que enquanto os equipamentos numerados de 1 a 8, de maior capacidade de processamento computaram 30 tarefas, enquanto aqueles numerados de 9 a 12, de menor capacidade computaram apenas 3 tarefas cada, ou seja, apenas 10 % dos demais.

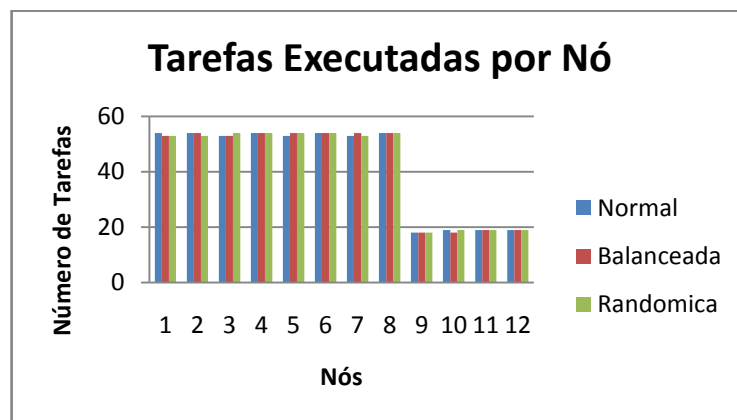


Figura 4.1: *Número de Tarefas executadas por nó, para uma base NR, em 12 máquinas, sendo 8 do tipo 1 e 4 do tipo 2, para consulta com 1000 seqüências e 500 aminoácidos cada.*

É mostrado ainda que o tempo total gasto para a execução das tarefas utilizando o particionamento normal foi de aproximadamente 52 minutos, enquanto no particionamento randômico foi de 49 minutos e no balanceado foi de 48 minutos, cerca de 5 % inferior ao maior tempo, ou seja, os diferentes tipos de particionamento utilizados para a execução não influenciaram, de forma significativa, o tempo total de execução.

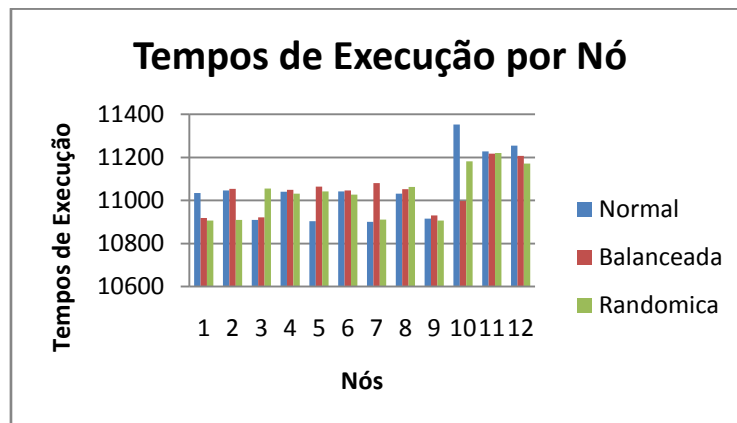


Figura 4.2: Tempo de Execução das tarefas designadas a cada nó, para uma base NR, em 12 máquinas, sendo 8 do tipo 1 e 4 do tipo 2, para consulta com 1000 seqüências e 500 bases cada.

A Figura 4.3 mostra graficamente que as máquinas de mesma capacidade de processamento computaram um número semelhante de tarefas, sendo que aquelas de maior capacidade computaram 30 tarefas, enquanto as de menor capacidade computaram apenas 10% deste total, ou seja, apenas 3 tarefas em cada nó. A Figura 4.4 mostra que, apesar de terem processado menos tarefas, os tempos de execução dos nós de menor capacidade foram maiores que os demais.

SQ_RN_EC_500_500 - Normal: RN_NT_12_12_FSD_EC_500_500

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	2249.55	3169.45	2246.93	922.42	30
2	2226.75	3169.45	2225.39	943.97	30
3	2228.84	3169.36	2226.88	942.46	30
4	2223.70	3169.64	2221.98	947.56	30
5	2217.94	3169.42	2216.48	952.92	30
6	2216.97	3169.40	2214.98	954.42	30
7	2222.00	3169.52	2220.73	948.74	30
8	2237.73	3169.51	2236.18	933.28	30
9	2989.42	3169.68	2988.35	181.23	3
10	3152.65	3169.69	3152.03	17.59	3
11	2613.28	3169.74	2612.67	556.94	3
12	3168.70	3169.72	3168.13	1.50	3

FSD 3171.96 3168.43 1.00 2.41

SQ_RN_EC_500_500 - Balanceado: RN_NT_12_12_FSD_EC_500_500

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	2220.14	2933.31	2218.57	714.65	30
2	2224.16	2933.97	2222.80	711.05	30
3	2214.65	2933.00	2212.30	720.52	30
4	2221.32	2932.62	2219.48	713.08	30
5	2223.20	2932.06	2220.85	711.13	30
6	2218.39	2933.48	2216.58	716.87	30
7	2273.25	2932.18	2271.56	660.58	31
8	2228.29	2933.85	2226.42	707.36	30

9	2394.54	2932.23	2394.10	538.08	2
10	2930.03	2931.12	2929.45	1.57	3
11	2823.85	2933.00	2823.01	109.88	3
12	2613.92	2932.96	2613.29	319.62	3

FSD 2935.20 2931.50 1.09 2.50

SQ_RN_EC_500_500 - Randomico: RN_NT_12_12_FSD_EC_500_500

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	2239.31	2982.30	2236.85	745.38	30
2	2234.76	2982.19	2232.62	749.56	30
3	2211.31	2982.09	2209.66	772.36	30
4	2179.50	2982.36	2178.31	804.04	30
5	2203.16	2982.16	2201.46	780.69	30
6	2219.22	2982.14	2217.58	764.53	30
7	2198.78	2982.25	2197.02	785.21	30
8	2217.12	2982.24	2215.59	766.61	30
9	2981.26	2982.41	2980.14	2.21	3
10	2710.08	2982.41	2709.44	272.89	3
11	2856.73	2982.45	2856.32	126.07	3
12	2920.17	2982.44	2919.66	62.73	3

FSD 2984.75 2981.03 1.14 2.47

Listagem 4.2: *Execução da Estratégia Sob Demanda para uma base NT, em 12 máquinas, sendo 8 do tipo 1 e 4 do tipo 2 para consulta com 500 seqüências e 500 bases nucleotídicas cada.*

As Figuras 4.5 e 4.6 apresentam a quantidade de leituras e gravações em disco efetuadas durante a execução do balaBLAST.

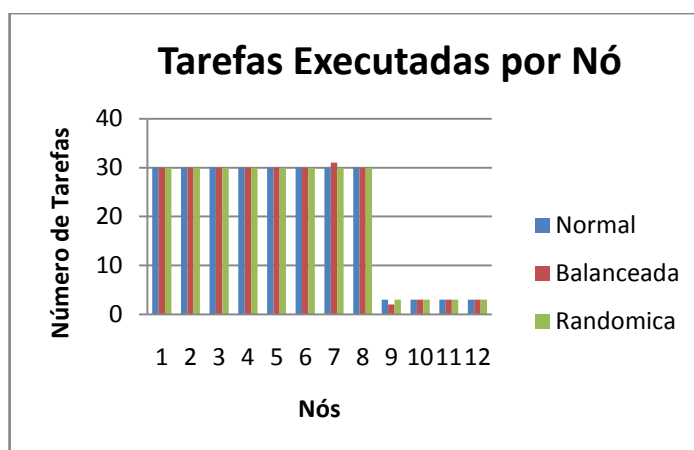


Figura 4.3: *Número de Tarefas executadas por nó, para uma base NT, em 12 máquinas da PUC-Rio, sendo 8 do tipo 1 e 4 do tpo 2, para consulta com 500 seqüências e 500 bases nucleotídicas cada.*

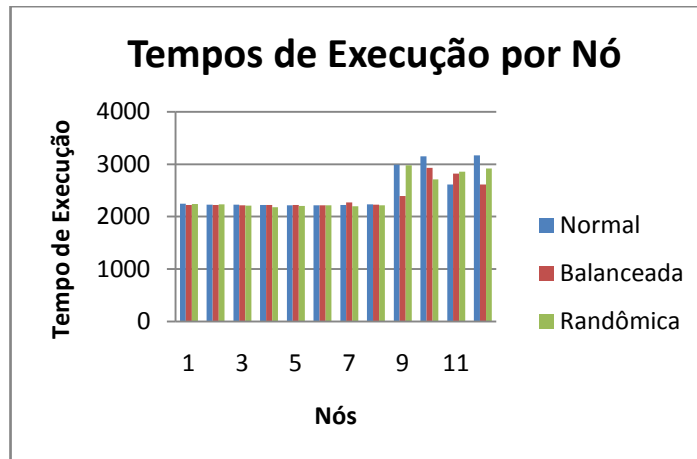


Figura 4.4: *Tempo de Execução das tarefas designadas a cada nó, para uma base NT, em 12 máquinas da PUC-Rio, sendo 8 do tipo 1 e 4 do tipo 2, para consulta com 500 seqüências e 500 bases nucleotídicas cada.*

Diferenças de tamanho de memória RAM, existentes entre os equipamentos envolvidos no teste, contribuíram para que houvesse um maior esforço de leitura de fragmentos da base de dados em equipamentos com menor capacidade de processamento.

A Figura 4.5 mostra que os nós de 9 a 12 efetuaram um número maior de leituras que os demais. Do total de leituras efetuadas durante a execução, cerca de 90% foram efetuadas por estes nós, o que demonstra que houve um maior esforço computacional por parte dos equipamentos de menor capacidade de processamento na execução de cada tarefa, em relação aos demais. Tal fato explica o porquê do balaBLAST ter atribuído mais tarefas aos equipamentos de maior capacidade, conforme mostrado na figura 4.3, no sentido de promover um equilíbrio da carga, e demonstra que a estratégia de balanceamento do balaBLAST realmente funcionou.

O número de gravações efetuadas, mostradas na Figura 4.6, corresponde a 0,68% das atividades de E/S efetuadas durante a execução do balaBLAST, e relaciona-se ao esforço para a elaboração de relatórios parciais do BLAST em cada um dos equipamentos do cluster. No entanto, o número de gravações efetuadas não teve uma influência significativa sobre o tempo total de execução da ferramenta.

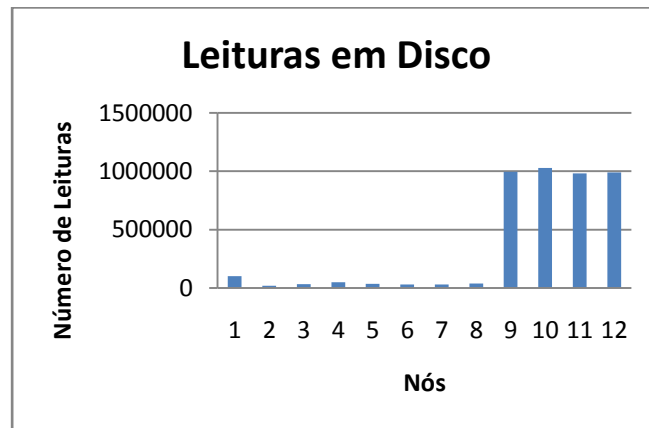


Figura 4.5: *Número de Leituras efetuadas em cada nó, durante a execução para uma base NT, em 12 máquinas da, sendo 8 do tipo 1 e 4 do tipo 2, para consulta com 500 seqüências e 500 bases nucleotídicas cada.*

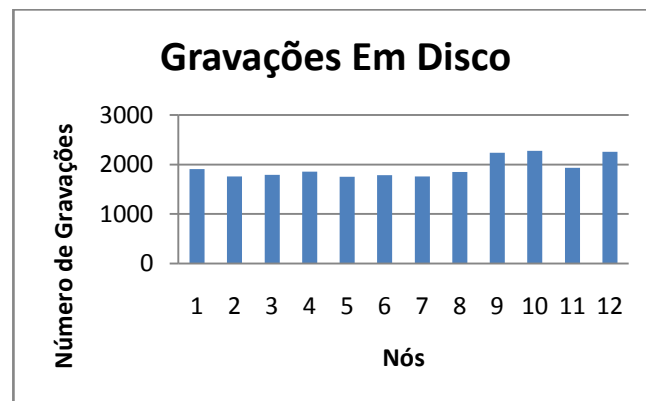


Figura 4.6: *Número de Gravações efetuadas em cada nó, durante a execução para uma base NT, em 12 máquinas, sendo 8 do tipo 1 e 4 do tipo 2, para consulta com 500 seqüências e 500 bases nucleotídicas cada.*

4.3.2

Testes utilizando Somente Equipamentos do Cluster da PUC-Rio – Equipamentos Homogêneos

A Listagem 4.3 corresponde a uma execução efetuada em 8 máquinas do *cluster* da PUC-Rio, para a base de dados NR, com as mesmas capacidades de processamento, todas do tipo 1, considerando 500 seqüências de consulta com aproximadamente 500 aminoácidos cada. Ela mostra que todas as máquinas processaram 21 tarefas, ou seja, o mesmo número de tarefas, mas não o mesmo

número de seqüências, e que os tempos de processamento foram semelhantes, algo em torno de 1 hora e 30 minutos, com uma diferença percentual que não ultrapassa 0,72%, o que significa que o algoritmo de balanceamento funcionou de forma satisfatória. A Figura 4.7 mostra graficamente que os 8 nós considerados no teste processaram 21 tarefas cada um.

Para este teste foi considerado somente o particionamento normal da base de dados, tendo em vista que os demais tipos de particionamento nada acrescentam em termos de desempenho na execução do balaBLAST, como pôde ser observado nos testes efetuados na Seção 4.3.1.

SQ_RP_NR_500_500 - RP_NR_8_8_FSD_NR_500_500: MPI PUC

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	5706,63	5761,90	5694,62	67,28	21
2	5681,34	5762,26	5667,15	95,10	21
3	5673,32	5762,63	5658,50	104,13	21
4	5698,66	5761,64	5685,91	75,72	21
5	5674,40	5762,15	5661,62	100,53	21
6	5665,41	5762,29	5651,72	110,57	21
7	5700,52	5762,73	5686,60	76,13	21
8	5684,39	5762,18	5668,34	93,83	21

FragSobdemanda - 5772,39 - 5706,99 - 55,29 - 10,03

Listagem 4.3: *Execução da Estratégia Sob Demanda para uma base NR, em 8 máquinas do tipo 1, para consulta com 500 seqüências e 500 aminoácidos cada.*



Figura 4.7: *Número de Tarefas executadas por nó, para uma base NR, em 8 máquinas da PUC-Rio, com igual capacidade de processamento, para consulta com 500 seqüências e 500 aminoácidos cada.*

A Figura 4.8 mostra os tempos de execução das tarefas em cada nó. Apesar do tempo de execução ter sido maior no nó 1 e menor no nó 6, tal fato não significa, necessariamente, que algum deles tenha processado um número maior, ou menor, de seqüências (não tarefas), mas indica que a estratégia de balanceamento foi capaz de gerir as diferentes quantidades de processamento entre os nós, e efetuar, de forma coerente, a distribuição de carga.

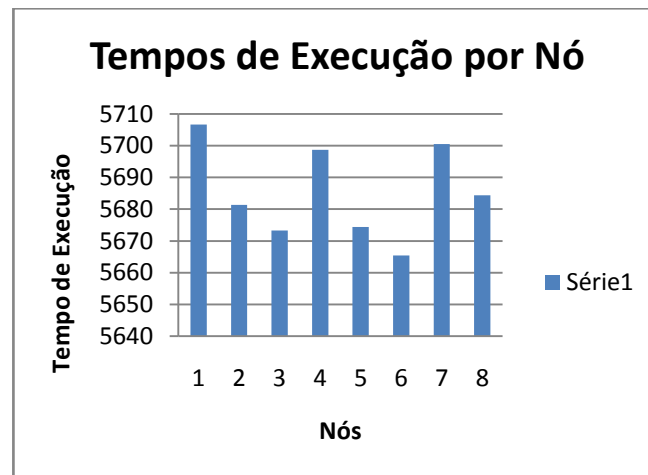


Figura 4.8: *Tempo de Execução das tarefas designadas a cada nó, para uma base NR, em 8 máquinas da PUC-Rio, com igual capacidade de processamento, para consulta com 500 seqüências e 500 aminoácidos cada.*

4.3.3

Teste utilizando o Grid

A Listagem 4.4 apresenta dados estatísticos sobre a execução do balaBLAST, em um ambiente de grid. Para este teste foram utilizadas 8 máquinas, sendo 4 máquinas, do tipo 1, pertencentes à PUC-Rio, numeradas de 1 a 4, e 4 máquinas, do tipo 3, pertencentes à UFF, numeradas de 5 a 8, considerando 500 seqüências retiradas da base NR com aproximadamente 500 aminoácidos cada uma, e analisadas contra a própria base NR.

SQ_RP_NR_500_500 - RP_NR_8_8_FSD_NR_500_500: Grid como Grid , , , 4 X 4

No	Conclusao	Morte	Blast	Comunica	Tarefas
1	6043,52	6130,07	6029,99	100,07	22
2	6035,90	6130,12	6023,00	107,12	22
3	6057,54	6130,17	6043,00	87,17	22
4	6045,50	6130,10	6031,45	98,65	22

5	6072,23	6130,48	6057,53	70,97	20
6	5988,82	6130,68	5974,48	155,61	20
7	5997,35	6130,42	5983,31	145,67	20
8	6014,41	6130,45	5999,34	129,15	20

FragSobdemanda - 6141,33 - 6071,76 - 58,40 - 11,12

Listagem 4.4: *Execução da Estratégia Sob Demanda para uma base NR, em um ambiente de grid, utilizando 8 máquinas, sendo 4 do tipo 1 da PUC-Rio, e 4 do tipo 3 da UFF, para consulta com 500 seqüências e 500 aminoácidos cada.*

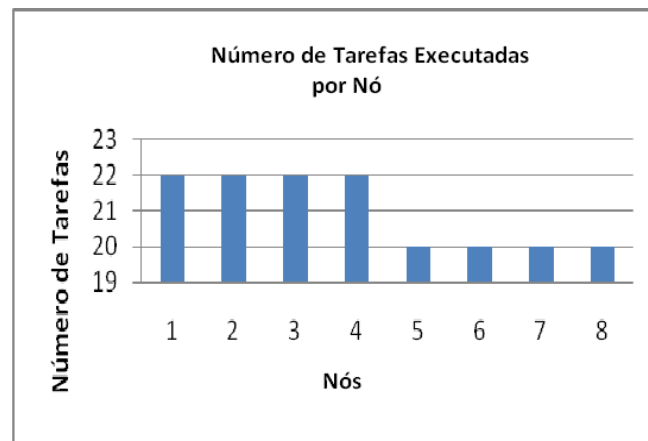


Figura 4.9: *Número de Tarefas executadas, para uma base NR em um ambiente de grid, utilizando 8 máquinas, sendo 4 do tipo 1 da PUC-Rio e 4 do tipo 3 da UFF, para consulta com 500 seqüências e 500 aminoácidos cada.*

A Figura 4.9 mostra que, embora possuindo maior capacidade de processamento, os equipamentos pertencentes à UFF processaram 20 tarefas, enquanto que os da PUC-Rio processaram 22. No entanto, a Figura 4.10 mostra que os tempos de execução das máquinas da UFF, em média, foram menores do que os da PUC. Apesar da aparente inconsistência, o fato pode ser explicado observando a coluna “Comunica” da Listagem 4.4, graficamente representada na Figura 4.11, que mostra que o tempo de comunicação do nó mestre, situado na PUC-Rio, com os equipamentos da UFF foram, em média, 27% mais altos e pelo fato de um ambiente de grid não ser controlado, e portanto com possibilidade de existirem outras aplicações sendo executadas no momento do teste.

Como as tarefas processadas pelos nós de mesma capacidade de processamento foram os mesmos, e o tempo de conclusão das tarefas em cada um dos nós foi aproximadamente o mesmo, conforme apresentado nas Figuras 4.9 e

4.10, pode-se concluir que o balaBLAST efetuou com sucesso o balanceamento de carga durante a execução do teste.

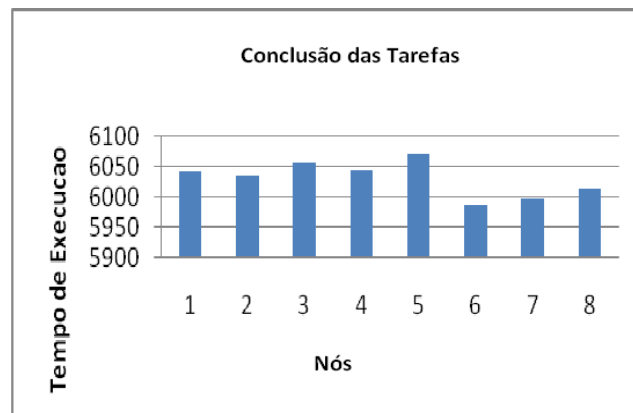


Figura 4.10: Tempo de Execução, das tarefas designadas a cada nó, para uma base NR em um ambiente de grid, utilizando 8 máquinas, sendo 4 da PUC-Rio e 4 da UFF, para consulta com 500 seqüências e 500 aminoácidos cada.

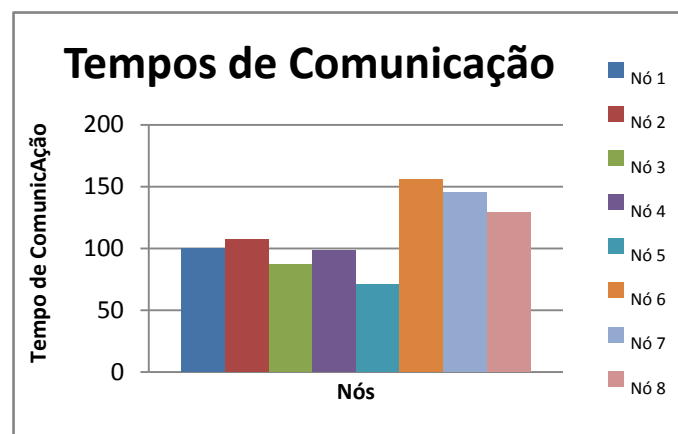


Fig.4.11: Tempos de Comunicação entre o nó mestre situado na PUC-Rio e os nós trabalhadores durante a execução do balaBLAST.

4.4

Conclusão

Pelos resultados obtidos a partir da aplicação dos testes descritos neste capítulo, além de outros que consideraram outras condições de execução do balaBLAST, é possível afirmar que a ferramenta incorpora, de fato, uma estratégia eficaz de balanceamento de carga, que funciona tanto em um ambiente de *cluster* quanto de *grid*.

Contudo, tal fato não implica que o balaBLAST seja capaz de prover o melhor desempenho. Na realidade, como foi citado anteriormente, as características intrínsecas das consultas é que determinam o esforço computacional necessário ao processamento. Assim, diferentes ordenações das seqüências de consulta podem implicar em diferentes esforços para execução da ferramenta, mas apenas uma delas será ótima para aquela estratégia de balanceamento.