



Daniel Xavier de Sousa

**Estratégias de Balanceamento de Carga para
Avaliação Paralela do BLAST com Bases de
Dados Replicadas e Fragmentos Primários**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática da PUC-Rio

Orientador: Prof. Sérgio Lifschitz

Rio de Janeiro
Julho de 2007



Daniel Xavier de Sousa

**Estratégias de Balanceamento de Carga para
Avaliação Paralela do BLAST com Bases de
Dados Replicadas e Fragmentos Primários**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Sérgio Lifschitz

Orientador

Departamento de Informática — PUC-Rio

Prof. Luiz Fernando Bessa Seibel

Departamento de Informática — PUC-Rio

Prof. Marta Lima de Queirós Mattoso

COPPE-UFRJ

Prof. Patrick Valduriez

INRIA, França

Prof. José Eugenio Leal

Coordenador Setorial do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 27 de Julho de 2007

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Daniel Xavier de Sousa

Possui graduação em Bacharelado em Ciência da Computação pela Universidade Católica de Goiás (2004). Tem experiência na área de Ciência da Computação, com ênfase em Banco de Dados, atuando principalmente nos seguintes temas: Bancos de Dados Distribuídos, Biologia Computacional e Processamento Paralelo. Natural de Goiânia, Goiás.

Ficha Catalográfica

Sousa, Daniel Xavier

Estratégias de Balanceamento de Carga para Avaliação Paralela do BLAST com Bases de Dados Replicadas e Fragmentos Primários / Daniel Xavier de Sousa; orientador: Sérgio Lifschitz. — Rio de Janeiro : PUC-Rio, Departamento de Informática, 2007.

v., 85 f: il. ; 29,7 cm

1. Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui referências bibliográficas.

1. Informática – Tese. 2. BLAST. 3. Balanceamento de Carga. 4. Bancos de Dados Distribuídos. 5. Processamento Paralelo.

I. Lifschitz, Sérgio. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Dedico este trabalho aos meus pais e aos meus irmãos, com os quais
compartilho minhas alegrias.

Agradecimentos

Inicialmente a Deus, que, como Pai, sempre esteve ao meu lado, nos bons e maus momentos, mostrando principalmente que não há vitória se acaso as conquistas conduzirem à perda de mim mesmo.

À minha família, que sempre me motivou, ensinando a importância de não desistir, incentivando para que tudo fosse feito com amor.

Ao meu orientador, Sérgio Lifschitz, que apostou em mim, dando credibilidade e confiança. Obrigado professor, pelas palavras de incentivo e pelo companheirismo.

Ao amigo e professor Wellington Martins, que foi meu primeiro colaborador para que este trabalho ocorresse.

Salve, amigos da República "Les Miserables": com vocês experimentei ótimos momentos. Agradeço imensamente o aprendizado que cada um repasou, igualmente os chopes, baladas e o companheirismo.

Aos amigos do grupo da pastoral PUC-Rio, que foi o peso espiritual necessário para que tudo ocorresse da melhor forma. Com vocês, tive a oportunidade de rezar, conversar e rir muito, os levarei por toda a minha vida.

Também de forma muito especial, às amigas Cibele e Dani e aos amigos Cristian, Fabrício, Herbert Duarte, Il, Leiser, e William. Agradeço por participarem da minha vida.

Ao grupo Bioinfo-PUC-Rio e ao pessoal da FIOCRUZ (Antônio, Catanho e Thomas), que compartilharam os momentos de discussões e novas idéias.

Aos professores e colegas de mestrado, pelo convívio no decorrer das aulas. Em especial ao professor Hermann e a todos do suporte, que de forma atenciosa atendiam às nossas solicitações: salve Zé, Anderson e Marcos.

Um enorme agradecimento ao João Cândido Portinari, que além de me receber em seu projeto, proporcionou ótimos diálogos.

Às instituições CAPES, VRAc (PUC-Rio) e ao Projeto Portinari, pelo apoio financeiro.

À banca, pela oportunidade de correção do meu trabalho. Saibam que é uma honra.

Por último agradeço as participações do nosso mascote, o inestimável gambá, que, com o ar da graça, se fez presente. Gambazinho, saudades!

Resumo

Sousa, Daniel Xavier; Lifschitz, Sérgio. **Estratégias de Balanceamento de Carga para Avaliação Paralela do BLAST com Bases de Dados Replicadas e Fragmentos Primários**. Rio de Janeiro, 2007. 85p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Na área de biologia computacional a busca por informações relevantes em meio a volumes de dados cada vez maiores é uma atividade fundamental. Dentre outras, uma tarefa importante é a execução da ferramenta BLAST (Basic Local Alignment Search Tool), que possibilita comparar biosseqüências a fim de se descobrir homologias entre elas e inferir as demais informações pertinentes. Um dos problemas a serem resolvidos no que diz respeito ao custo de execução do BLAST se refere ao tamanho da base de dados, que vem aumentando consideravelmente nos últimos anos. Avaliar o BLAST com estratégias paralelas e distribuídas com apoio de agrupamento de computadores tem sido uma das estratégias mais utilizadas para obter ganhos de desempenho. Nesta dissertação, é realizada uma alocação física replicada da base de dados (de seqüências), onde cada réplica é fragmentada em partes distintas, algumas delas escolhidas como primárias. Dessa forma, é possível mostrar que se aproveitam as principais vantagens das estratégias de execução sobre bases replicadas e fragmentadas convencionais, unindo flexibilidade e paralelismo de E/S. Associada a essa alocação particular da base, são sugeridas duas formas de balanceamento dinâmico da carga de trabalho. As abordagens propostas são realizadas de maneira não intrusiva no código BLAST. São efetuados testes de desempenho variados que demonstram não somente a eficácia no equilíbrio de carga como também eficiência no processamento como um todo.

Palavras-chave

BLAST. Balanceamento de Carga. Bancos de Dados Distribuídos. Processamento Paralelo.

Abstract

Sousa, Daniel Xavier; Lifschitz, Sérgio. **Workload Balancing Strategies for Parallel BLAST evaluation on Replicated Databases and Primary Fragments**. Rio de Janeiro, 2007. 85p.
MsC Thesis — Department of Informática, Pontifícia Universidade Católica do Rio de Janeiro.

A fundamental task in the area of computational biology is the search for relevant information within the large amount of available data. Among others, it is important to run tools such as BLAST - Basic Local Alignment Search Tool - efficiently, which enables the comparison of biological sequences and discovery of homologies and other related information. However, the execution cost of BLAST is highly dependent on the database size, which has considerably increased. The evaluation of BLAST in distributed and parallel environments like PC clusters has been largely investigated in order to obtain better performances. This work reports a replicated allocation of the (sequences) database where each copy is also physically fragmented, with some fragments assigned as primary. This way we show that it is possible to execute BLAST with some nice characteristics of both replicated and fragmented conventional strategies, like flexibility and I/O parallelism. We propose two dynamic workload balancing strategies associated with this data allocation. We have adopted a non-intrusive approach, i.e., the BLAST code remains unchanged. These methods are implemented and practical results show that we achieve not only a balanced workload but also very good performances.

Keywords

BLAST. Workload Balancing. Distributed Database. Parallel Processing.

Sumário

1	Introdução	12
2	Descrição do Problema	16
2.1	Desbalanceamento de Carga	18
2.2	BLAST com bases fragmentadas	22
2.3	Estratégias sobre bases em processo de atualização	25
2.4	Robustez do sistema	26
2.5	Conclusão	26
3	Similaridade e tamanho da seqüência de consulta no BLAST	28
3.1	Fatores de desbalanceamento do BLAST	29
3.2	Trabalhos relacionados à similaridade	30
3.3	Testes experimentais	31
3.4	Conclusão	38
4	Avaliação paralela do BLAST	40
4.1	Estratégia com Replicação Total	41
4.2	Estratégia Fragmentada	43
4.3	Conclusão	48
5	Soluções Propostas	49
5.1	Estratégia Sob Demanda	52
5.2	Estratégia Corretiva	62
5.3	Conclusão	67
6	Avaliação prática das novas Estratégias	69
6.1	Estado de Aceleração	69
6.2	Diferentes tamanhos da Base de Dados	70
6.3	Diferentes Números de Fragmentos da Base de Dados	75
6.4	Conclusão	75
7	Conclusão e trabalhos futuros	77
7.1	Resumo	77
7.2	Contribuições	78
7.3	Trabalhos Futuros	79
A	Descrição do Agrupamento de Computadores e Programas utilizados	85

Lista de figuras

2.1	<i>Tendência de crescimento dos banco de dados de seqüências (http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html)</i>	22
3.1	<i>Execução do BLASTN para seqüências aleatórias e similares utilizando uma base de dados menor que a memória RAM.</i>	33
3.2	<i>Execução do BLASTP para seqüências aleatórias e similares utilizando uma base de dados menor que a memória RAM.</i>	33
3.3	<i>Execução do BLASTN para seqüências aleatórias e similares utilizando uma base de dados maior que a memória RAM.</i>	34
3.4	<i>Execução do BLASTP para seqüências aleatórias e similares utilizando uma base de dados maior que a memória RAM.</i>	35
3.5	<i>Número de extensões obtidas com sucesso pelo BLASTN para seqüências aleatórias e seqüências similares.</i>	36
3.6	<i>Número de extensões obtidas com sucesso pelo BLASTP para seqüências aleatórias e seqüências similares.</i>	37
3.7	<i>Execução Paralela para estratégia Fragmenta usando seqüências de consulta aleatórias.</i>	37
3.8	<i>Execução Paralela para estratégia Fragmenta usando seqüências de consulta similares.</i>	38
5.1	<i>Exemplo da alocação da base de dados com replicação total com fragmentos para três máquinas. Em cada máquina de trabalho o banco inteiro está fragmentado em distintas partições, e em destaque estão os fragmentos primários para cada estação.</i>	51
5.2	<i>Algoritmo de funcionamento da estratégia Sob Demanda.</i>	54
5.3	<i>Balanceamento de carga utilizando $rS=2$.</i>	56
5.4	<i>Balanceamento de carga utilizando $rS=32$.</i>	56
5.5	<i>Balanceamento de carga utilizando $rS=64$.</i>	57
5.6	<i>Balanceamento de carga utilizando $rS=128$.</i>	57
5.7	<i>Balanceamento de carga utilizando $rS=256$.</i>	57
5.8	<i>Tempos de execução para estratégia Sob Demanda, variando os valores de rS. Para todos os tempos foram utilizadas as mesmas 1.000 seqüências de consulta e a base nt particionado em 24 fragmentos.</i>	59
5.9	<i>Apresenta o resultado da execução de várias estratégias e o desempenho da Estratégia Sob Demanda, mostrando a porcentagem em tempo de processamento em relação à estratégia Serial. Em todas as estratégias foram utilizadas as mesmas seqüências de consulta e o mesmo banco nt.</i>	62
5.10	<i>Fluxograma de funcionamento da estratégia Corretiva</i>	64
5.11	<i>Balanceamento de carga obtido pela Estratégia Corretiva.</i>	66
5.12	<i>Compara a Estratégia Corretiva em relação à varias outras, mostrando a porcentagem em tempo de processamento em relação à estratégia Serial. Em todas as estratégias foram utilizadas as mesmas seqüências de consulta e o mesmo banco nt.</i>	67

6.1	<i>Execução da estratégia Corretiva variando o número de máquinas de trabalho.</i>	70
6.2	<i>Execução da estratégia Sob Demanda variando o número de máquinas de trabalho, com $rS=32$.</i>	71
6.3	<i>Execução das estratégias utilizando a base de dados ecoli.nt.</i>	73
6.4	<i>Execução das estratégias utilizando a base de dados SwissProt</i>	74
6.5	<i>Execução das estratégias utilizando a base de dados nt.</i>	74
6.6	<i>Execução das Estratégias Corretiva e Sob Demanda alterando o número de fragmentos que compõem a base de dados.</i>	76

Lista de tabelas

- 2.1 *Distintas execuções do BLAST utilizando diferentes seqüências de consulta para cada execução e duas opções de banco de dados.* 17
- 6.1 *Propriedade das bases de dados utilizadas nos testes.* 71