Análise de Desempenho e Memória do Programa de Alinhamento Pa-Star no Supercomputador Santos Dumont

Kelen Souza[1,2], Micaella Coelho[1], Carla Osthoff[1], Kary Ocaña[1]

[1]Laboratório Nacional de Iniciação Científica (LNCC) – RJ/Brasil

[2]Faculdade de Educação Tecnológica do Rio de Janeiro (FAETERJ - Petrópolis) – RJ/Brasil

{kelenbs, micaella, osthoff, karyann}@lncc.br

Abstract. *This study evaluated the performance of the bioinformatics program PA-Star, used for multiple sequence alignment, on two architectures of the Santos Dumont supercomputer: the Mesca2, with higher computational and memory capacity, and the Intel Xeon Ivy Bridge node. The biological sequences processed varied in size, quantity, and level of similarity. It was observed that Mesca2 showed superior performance for files with high RAM demand, while the Intel Xeon Ivy Bridge was more efficient for files with lower demand. These results suggest that Mesca2 is essential for processing alignments with high RAM consumption on Santos Dumont.*

Resumo. *Este estudo avaliou o desempenho do programa de bioinformática PA-Star, utilizado para alinhamento múltiplo de sequências, em duas arquiteturas do supercomputador Santos Dumont: o Mesca2, com maior capacidade computacional e de memória, e o nó Intel Xeon Ivy Bridge. As sequências biológicas processadas variaram em tamanho, quantidade e nível de similaridade. Observou-se que o Mesca2 apresentou desempenho superior para arquivos com alta demanda de RAM, enquanto o Intel Xeon Ivy Bridge foi mais eficiente para arquivos com menor demanda. Esses resultados sugerem que o Mesca2 é essencial para o processamento de alinhamentos com elevado consumo de memória RAM no Santos Dumont.*

1. Introdução

O alinhamento múltiplo de sequências (AMS) é uma técnica fundamental na bioinformática, utilizada para alinhar três ou mais sequências biológicas, como DNA, RNA ou proteínas, a fim de identificar regiões de semelhança que possam indicar relações funcionais, estruturais ou evolutivas [1]. No entanto, a execução do AMS em CPUs apresenta desafios significativos, principalmente com sequências longas e complexas ou grandes quantidades de dados [2].

Neste estudo, utilizamos o PA-Star [3], um programa baseado no algoritmo de busca A-star (A*), que identifica o caminho de custo mínimo entre dois pontos em um grafo direcionado e ponderado. Embora o A* seja aplicável a uma ampla gama de problemas de busca de caminho, o foco deste trabalho é a implementação da busca paralela A-Star no PA-Star para resolver o problema NP-difícil do AMS.

O objetivo deste estudo é investigar o desempenho do PA-Star em diversos ambientes computacionais e configurações, principalmente em relação à demanda de memória RAM, que é seu maior desafio. Os testes foram realizados no supercomputador Santos Dumont (SDumont, https://sdumont.lncc.br/), em duas arquiteturas de memória compartilhada: **Mesca2**, que se destaca pela alta capacidade de memória RAM e desempenho computacional, e o nó **Intel Xeon Ivy Bridge**, que possui uma quantidade de memória adequada para a maioria dos testes e é uma máquina mais recente. O PA-Star, voltado para AMS em CPUs, foi desenvolvido por Daniel Sundfeld (UnB) [3].

2. Trabalho Relacionado

O AMS é um problema NP-completo, geralmente resolvido com algoritmos heurísticos, embora soluções exatas sejam preferíveis para um número limitado de sequências. Sundfeld et al. (2018) [2,3] propuseram o PA-Star, um algoritmo *multithreaded* baseado no A-Star, projetado para rodar em CPU, utilizando RAM e disco para gerenciar o alto consumo de memória no alinhamento. O PA-Star aplica uma política sensível à localidade para alocação de trabalho às *threads*, otimizando os recursos da CPU [2,3]. Neste trabalho, testamos o programa pela primeira vez no supercomputador SDumont, comparando o desempenho na arquitetura Mesca2 e no nó Intel Xeon Ivy Bridge.

3. Metodologia

O estudo utilizou sequências de aminoácidos (proteínas) como dados de entrada, em arquivos no formato FASTA do repositório astar-msa no GitHub (https://github.com/danielsundfeld/astar_msa), mantido pelo criador do programa, Sundfeld et al. [3]. Seguindo a recomendação de Sundfeld, quatro arquivos iniciais (*glg, 1sbp, 1aboA, 1ac5*) foram selecionados como ponto de partida para os testes, com o objetivo de avaliar o comportamento do programa, sua escalabilidade e configuração de *threads*. Um segundo grupo, também indicado por Sundfeld, incluiu arquivos que, neste trabalho, serão chamados de intermediários (*gal4, 1gpb, arp, 1sesA, 2myr, 2cba, 1hvA, 2ack* e *actin*), e dentre esses alguns apresentam uma maior demanda de RAM para o alinhamento. A variação na demanda de RAM para o processamento deve-se às diferenças nas sequências de aminoácidos em cada arquivo, como quantidade e posição de gaps, fatores que impactam o tempo de alinhamento. Os testes no SDumont utilizaram duas arquiteturas de memória compartilhada: o Mesca2, com 240 núcleos e 6 TB de RAM (1, 12, 24, 48, 60, 120 e 240 *threads*), e o Intel Xeon Ivy Bridge de 2,4 GHz, com 64 GB de RAM e 24 núcleos (1, 12 e 24 *threads*).

4. Resultados

A Figura 1 apresenta os testes realizados com os arquivos iniciais no Mesca2, onde o melhor desempenho em termos de tempo de execução foi obtido com 24 *threads*. Cada configuração foi testada três vezes e, como não houve variação significativa, foi utilizada a média desses valores. A curva em vermelho indica que o consumo de RAM aumentou proporcionalmente ao número de *threads*, com 24 *threads* sendo ideais em termos de desempenho e tempo de execução.

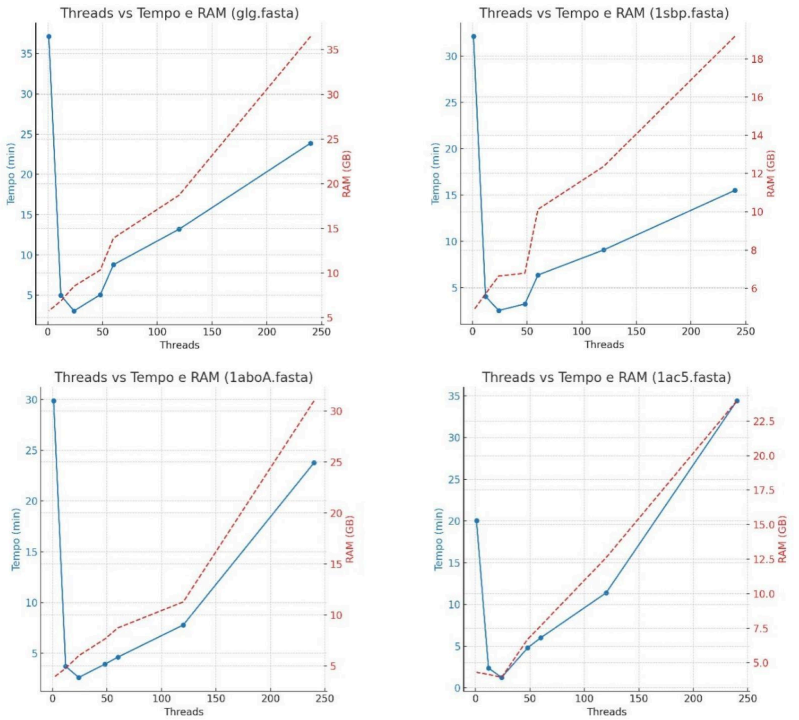

Fig. 1. Tempo de execução e uso de RAM dos arquivos iniciais no Mesca2 para a observação do funcionamento do programa PA-Star. Curva vermelha: *Threads* vs. RAM; Curva azul: *Threads* vs. Tempo.

Posteriormente, os testes realizados no nó Intel Xeon Ivy Bridge revelaram que a melhor configuração foi com 24 *threads*. Assim, a Tabela 1 apresenta os dados obtidos a partir dos testes com 24 *threads* para comparação. A tabela exibe as principais informações dos arquivos testados, incluindo os arquivos iniciais (os quatro primeiros) e intermediários (a partir do quinto). Ela apresenta o tamanho dos arquivos em kilobytes, o número de sequências (variando de 4 a 5), o tamanho da menor e da maior sequência de cada arquivo e a similaridade pós-alinhamento das sequências, realizada pelo PA-Star (a similaridade foi a mesma em ambas as arquiteturas). As colunas em azul mostram os dados referentes ao tempo de execução e uso de RAM para o AMS utilizando o Mesca2, enquanto as colunas em laranja exibem os dados obtidos no nó Intel Xeon Ivy Bridge.

A Tabela 1 indica que, entre os arquivos intermediários, *gal4* e *1gpb* não apresentam resultados nas colunas em laranja (relacionadas ao nó Intel Xeon Ivy Bridge), pois não puderam ser executados nesse nó. Isso evidencia que o Intel Xeon Ivy Bridge não atende aos requisitos de RAM necessários para o processamento de arquivos que exigem mais do programa, embora tenha mostrado bom desempenho nos arquivos iniciais e em alguns do grupo intermediário (destacados em negrito). Em contraste, o Mesca2 (colunas em azul) se destacou no processamento dos arquivos que exigem mais RAM (destacados em negrito), demonstrando sua superior capacidade de processamento.

Arquivos (tipo .fasta)	Tamanho do Arquivo	Nº de Sequências	Menor Sequência	Maior Sequência	Similaridade (após alinhamento)	Tempo (Mesca)	Tempo (Ivy Bridge)	RAM (Mesca)	RAM (Ivy Bridge)
glg	2.4K	5	438	486	26.80%	00h:02m:57s	**00h:02m:28s**	8.13G	**6.83G**
1sbp	1.3K	5	224	263	12.36%	00h:02m:43s	**00h:02m:05s**	6.68G	**5.63G**
1aboA	335	5	49	80	28.75%	00h:02m:41s	**00h:02m:08s**	5.98G	**4.95G**
1ac5	1.8K	4	421	483	25.10%	00h:01m:46s	**00h:01m:23s**	4,24G	**3.82G**
gal4	1.9K	5	335	395	15.80%	**16h:00m:48s**	-	**434.96G**	-
1gpb	4.0K	5	796	828	42.60%	**03h:46m:21s**	-	**200.95G**	-
arp	2.1K	5	380	418	24.16%	**00h:28m:00s**	00h:29m:38s	**48.85G**	49.28G
1sesA	2.2K	5	417	442	29.87%	**00h:14m:48s**	00h:17m:43s	**31.70G**	34.15G
2myr	1.7K	4	340	474	14.94%	00h:08m:42s	**00h:07m:24s**	21.90G	**19.40G**
2cba	1.3K	5	237	259	22.15%	00h:06m:11s	**00h:04m:47s**	13.69G	**11.88G**
1hvA	928	5	136	199	14.07%	00h:04m:35s	**00h:04m:18s**	10.31G	**9.80G**
2ack	2.4K	5	452	482	18.82%	00h:03m:12s	**00h:02m:39s**	8.96G	**8.11G**
actin	2.0K	5	379	395	40.25%	00h:03m:40s	**00h:02m:11s**	10.34G	**7.30G**

Tabela. 1. Informações dos arquivos e resultados nas arquiteturas Mesca2 e Ivy Bridge, usando 24 *threads*.

5. Conclusão

Os testes realizados no supercomputador SDumont com o PA-Star mostraram que o Mesca2 se destacou no processamento de arquivos com alta demanda de memória, como *gal4* e *1gpb*, enquanto o Intel Xeon Ivy Bridge apresentou desempenho superior apenas em arquivos com menor demanda. Arquivos que necessitavam de muita RAM, como *gal4* e *1gpb*, não puderam ser processados nesse nó. A superioridade do Mesca2 em cenários de alta exigência de memória ressalta sua importância para pesquisas futuras, especialmente considerando que apenas arquivos com 4 ou 5 sequências foram analisados até agora. Essa arquitetura permitirá testes com arquivos mais exigentes e um maior número de sequências, sendo essencial para a modelagem de workflows científicos que requerem alto poder de processamento e memória. Os próximos passos incluem a comparação do desempenho de diferentes arquiteturas, incorporando as atualizações em andamento no SDumont.

6. Referências

[1] Chowdhury B, Garai G (2017) A review on multiple sequence alignment from the perspective of genetic algorithm. Genomics 109:419–431. https://doi.org/10.1016/j.ygeno.2017.06.007

[2] Sundfeld, D. Alinhamento primário e secundário de sequências biológicas em arquiteturas de alto desempenho. 2017, 167 f., il. Tese (Doutorado em Informática). Universidade de Brasília, 2017.

[3] Sundfeld, D; Razzolini, C; Teodoro, G; Boukerche, A; Melo, ACM. PA-Star: A disk-assisted parallel A-Star strategy with locality-sensitive hash for multiple sequence alignment. Journal of Parallel and Distributed Computing, v. 112, p. 154-165, 2018.