

Avaliação de Desempenho de um Workflow Científico para Experimentos de RNA-Seq no Supercomputador Santos Dumont

Lucas Cruz¹, Micaella Coelho¹, Luiz Gadelha¹, Kary Ocaña¹, Carla Osthoff¹

¹Laboratório Nacional de Computação Científica (LNCC)
Petrópolis – RJ – Brasil

{lucruz, micaella, lgadelha, karyann, osthoff}@lncc.br

Abstract. *Large-scale scientific experiments are considered complex due to modeling of their activities, execution and big data analyses. In bioinformatics these experiments are modeling as scientific workflows using the High Performance Computing and data science concepts. This research paper presents the ParslRNA-Seq workflow for RNA-Seq experiments and analyses of performance execution on the Santos Dumont supercomputer using real data. The results show an improvement on performance, by comparing to execution on traditional way without parallelization and via Web, from 3 days to 11 hours, with reproducibility of biological results, by comparing ParslRNA-Seq to traditional or Web application execution. The workflow multithreading execution also indicates that the parametrization is dependent on Parsl and bowtie use.*

Resumo. *Experimentos científicos em larga escala são considerados complexos devido à modelagem de suas atividades, execução e análises de grandes volumes de dados. Na bioinformática esses experimentos são modelados como workflows científicos utilizando conceitos de computação de alto desempenho e ciência de dados. Neste artigo apresentamos o workflow ParslRNA-Seq para experimentos de RNA-Seq e análises de desempenho das execuções realizadas no supercomputador Santos Dumont usando dados reais. Os resultados mostram uma melhora no desempenho, quando comparado às execuções realizadas da forma tradicional sem paralelização e via Web, de 3 dias para 11 horas, com reproducibilidade de resultados de dados biológicos sensíveis. A execução multithreading do workflow indica também que a parametrização é dependente de Parsl e da atividade bowtie.*

1. Introdução

Com o surgimento de novas tecnologias nas áreas de sequenciamento biológico e biologia computacional, um grande volume de dados vem sendo produzido. A complexidade do processamento e análise desses dados traz um desafio para o meio computacional devido à necessidade de acoplar infraestruturas e tecnologias especializadas no âmbito de pesquisas em bioinformática. A tecnologia de Sequenciamento RNA (RNA-Seq) é utilizada na área de bioinformática no estudo do transcriptoma que é o conjunto de transcritos de uma célula em uma dada condição fisiológica ou de desenvolvimento. Experimentos de RNA-Seq tratam da análise de expressão diferencial de genes [Love et al. 2014] e se caracterizam por serem complexos, com longa duração, computacionalmente intensivos e pela produção de grande volume de dados.

Experimentos de RNA-Seq apresentam dependências entre as suas atividades e podem ser representados como workflows científicos [Cordeiro et al. 2013]. Um workflow científico é uma forma de representar, abstratamente, o encadeamento entre atividades dentro de um cenário experimental automatizando os processos experimentais, de modo computacional, e dando maior controle sobre eles [Braghetto and Cordeiro 2014]. Devido as características computacionais desse tipo de experimento é necessário utilizar ferramentas capazes de gerenciar a execução paralela e distribuída dos workflows em ambientes de Computação de Alto Desempenho (CAD) [Mattos et al. 2008].

É possível encontrar uma variedade de pipelines do workflow RNA-Seq já modeladas como, RSeqFlow e Wedring. No entanto, esses trabalhos mantêm como objetivo principal a análise de dados reais do experimento, e não no desempenho e otimização com a utilização de ambientes de CAD. Há também, sistemas de gerência baseados em web, que permitem a modelagem de workflows para serem executados de forma automática, utilizando para isso a plataforma do administrador, como o Galaxy¹. Contudo, a preocupação ainda reside em análises e não em desempenho.

O presente trabalho apresenta, então, a modelagem do workflow de RNA-Seq, denominado ParslRNA-Seq e as respectivas análises de desempenho em ambiente de supercomputação. Os estudos de desempenho visam determinar as melhores combinações de parametrização das atividades que compõem o workflow e os recursos oferecidos pelo Parsl para ambientes CAD, permitindo o uso racional e eficiente dos recursos computacionais e a diminuição do tempo computacional de execução.

2. Workflow Científico RNA-Seq

A seção 2.1 traz informações sobre as atividades do workflow e sua modelagem. A seguir, na seção 2.2, uma breve descrição sobre a ferramenta Parallel Scripting Library (Parsl) e como ela foi configurada para execução no supercomputador Santos Dumont² (SDumont). A seção 2.3 discorre sobre a infraestrutura de CAD e como seus recursos estão sendo utilizados. Por fim, a seção 2.4 traz breves informações sobre a disposição dos dados biológicos para execução do workflow.

2.1. Modelagem

O workflow possui três atividades principais, como mostrado na Figura 1, que são executadas por dependência: A primeira atividade deve mapear leituras curtas em genomas e é realizada por um software dedicado a tal, chamado bowtie2 [Langmead and Salzberg 2012]. Esse software faz, basicamente, a comparação de caractere para caractere em arquivos extremamente longos; a segunda atividade realiza a contagem das leituras de forma individual para todos os alinhamentos da atividade anterior, utilizando o software HTSeq [Anders et al. 2014] e a terceira, atividade DESeq2, aplica estatísticas sobre contagens para se inferir genes diferencialmente expressos, dado um número de condições experimentais [Love et al. 2014].

Há de se destacar também, ainda na modelagem, as atividades que necessitam de barreiras para o controle de sincronização. Como nesse caso a terceira atividade, DESeq2, que é de chamada única e utiliza todos os dados gerados pela atividade HTSeq, conforme a

¹Disponível em <https://galaxyproject.org/>. Último acesso: 14/08/2020.

²Disponível em <https://sdumont.lncc.br/>. Último acesso: 14/08/2020.

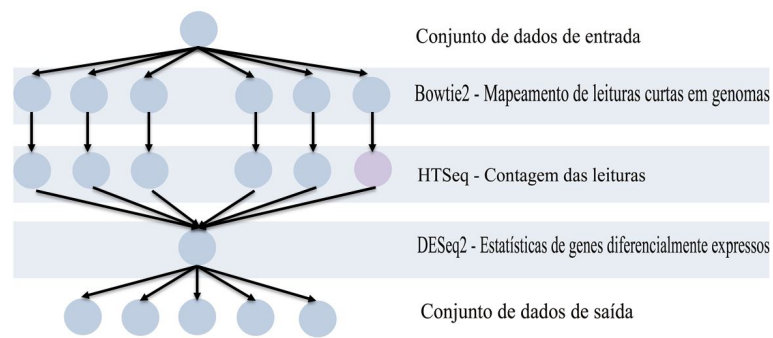


Figura 1. Modelagem Conceitual do Workflow RNA-Seq.

Figura 1. Isso sinaliza que o workflow ParslRNA-Seq pode executar de modo assíncrono as duas primeiras atividades, sendo que a última atividade é dependente da finalização da segunda atividade, HTSeq, pois consome e concatena todos os dados gerados. Por essa razão, a escolha da ferramenta para a gerência de execução paralela e distribuída é relevante no processo de desenvolvimento, pois ela será responsável por administrar a comunicação entre o workflow e o ambiente computacional no controle de sincronização.

2.2. Parsl - Parallel Scripting Library

Uma vez modelado o workflow é preciso que seja transcrito para representação computacional. O ParslRNA-Seq utiliza para isso o Parsl, que é uma ferramenta desenvolvida na linguagem de programação Python capaz de executar workflows em ambientes de CAD [Babuji et al. 2019]. O Parsl surge como uma tentativa de migrar as utilidades do Swift para uma linguagem rica em recursos, assim os usuários não necessitam de tempo demasiado para aprender a escrever workflows. Com o uso de decoradores, chamados Apps, é possível utilizar comandos externos diretamente do código Python. Além disso, o próprio Parsl irá fazer o controle de sincronização, havendo apenas necessidade que seja indicado onde devem haver as barreiras. Por fim, o motor de execução do Parsl contém diversas opções, permitindo execução em qualquer ambiente computacional. Esses fatos auxiliam muito na abstração das etapas do workflow, em termos de implementação e acoplagem a recursos computacionais.

Inicialmente será usado o ThreadPoolExecutor³, que é o modelo de execução utilizado pelo Parsl e dá suporte à execução multithreading utilizando recursos locais, a partir de um pool de threads que tem bifurcação do programa principal. Esse modelo permite a indicação máxima de threads que podem ser alocadas para serem utilizadas por uma aplicação, ou seja, o máximo de atividades que podem ser executadas de forma concorrente. Uma vez que a forma de implementação de um gerenciador de threads traz impacto direto no desempenho computacional, em termos de produtividade e redução na ocorrência de erros [Silva and Yokoyama 2011], é necessário o uso de um modelo desenvolvido e dedicado exclusivamente à gerência do ciclo de uso das threads, na execução de um workflow em ambientes de CAD.

2.3. O Ambiente Computacional Santos Dumont

O SDumont está entre as 500 máquinas mais poderosas do mundo. Possui uma capacidade de processamento de 5.1 Petaflop/s, com 34.688 CPUs multi-cores, distribuídas

³Disponível em <https://docs.python.org/3/library/concurrent.futures.html>. Último acesso: 14/08/2020.

em 1.132 nós computacionais que são interligados por uma rede de interconexão Infiniband FDR/HDR. Os nós computacionais possuem duas CPUs Ivy Bridge Intel Xeon E5-2695v2 (12c @2.4GHz) e 64Gb de memória RAM e uma GPU Nvidia K40. As análises de escalabilidade e desempenho do workflow foram realizadas usando os recursos computacionais no ambiente do SDumont, utilizando CPUs de um nó computacional que dispõe de 24 núcleos, por meio de um conjunto de cenários propostos e apresentados nas seguintes seções. Essas análises permitiram melhor abordar o problema de variabilidade de configuração das threads entre os softwares, a linguagem de gerência e o ambiente computacional. Posteriormente, será feita a implementação para o uso de mais nós computacionais com o modelo de execução do Parsl, HighThroughputExecutor.

2.4. Dados de Entrada

As análises de expressão gênica são realizadas dividindo os dados em grupos que representem uma dada condição apresentada da amostra, por exemplo, a evolução por tempo da diferenciação da expressão de genes envolvidos nas vias Wnt (Wingless pathway). As informações sobre o grupo a que cada arquivo pertence foram disponibilizadas pelo bioinformata e estão disponíveis no portal European Nucleotide Archive⁴ (ENA). Essas informações são de extrema importância para manter a integridade dos dados futuramente gerados e, até mesmo, para a validação dos resultados finais. Portanto, foi criado um arquivo no formato CSV (Comma-Separated Values) como metadado desses arquivos, que será utilizado quando requerido pelas atividades do workflow.

Os dados de entrada para o ParslRNA-Seq são o genoma de referência, o arquivo de características genômicas no formato GTF (Gene Transfer Format), os arquivos de sequenciamento no formato fastq e as condições experimentais da amostra. Esses dados de RNA-Seq se referem à via metabólica de sinalização transcricional Wnt, a que tem sido reportada com um papel regulatório no coração. Um total de seis arquivos de sequenciamento, divididos em dois grupos, três de controle e três de condição Wnt sequenciadas no *Mus Musculus* (rato) foram usados como dados de entrada pelo ParslRNA-Seq.

Há de se notar que o número de arquivos de sequenciamento e a divisão dos grupos podem variar conforme o experimento. Então, inicialmente, a análise de desempenho está ligada somente a esses seis arquivos, onde dois deles possuem o tamanho de 1.5GB e os outros 1.6GB, 1.7GB, 2.1GB e 2.6GB, totalizando 11GB. Posteriormente, alguns deles são replicados para fins de análises, como será observado na próxima seção.

3. Resultados e Análise

O mesmo experimento RNA-Seq foi modelado e executado no sistema Galaxy levando de 2 a 3 dias para finalizar a execução. Em comparação, com a utilização da ferramenta Parsl no SDumont o experimento demandou cerca de 11 horas, no pior dos cenários, como pode ser observado na Figura 2(a). Além do tempo de execução reduzido, os dados finais gerados foram validados pelo bioinformata.

Um outro ponto a ser considerado é que o software bowtie2, que compõe a primeira atividade do workflow, possui também recurso multithreading. O que torna a análise mais complexa. Então, foi realizada uma exploração do comportamento do workflow dentro de cenários sem e com a utilização dos recursos de paralelização do bowtie2,

⁴Disponível em <https://www.ebi.ac.uk/ena/browser/view/PRJNA382836>. Último acesso: 14/08/2020.

sob quatro experimentos: o primeiro é pela gerência paralelizada a partir das ferramentas do Parsl; o segundo, pela paralelização apenas da primeira atividade, Bowtie; o terceiro é fazendo utilização multithreading tanto pelo Parsl, quanto pelo bowtie2 e o quarto é pela variação da quantidade de arquivos de entrada. Também será feita uma análise do balanceamento de carga, utilizando para isso o perfilador VTune⁵. Todos os experimentos realizados, com exceção do último que é feito pela replicação dos arquivos de sequenciamento, fazem uso dos dados de entrada mencionados na subseção 2.4.

3.1. Análise de desempenho a partir do gerenciamento de threads pelo Parsl

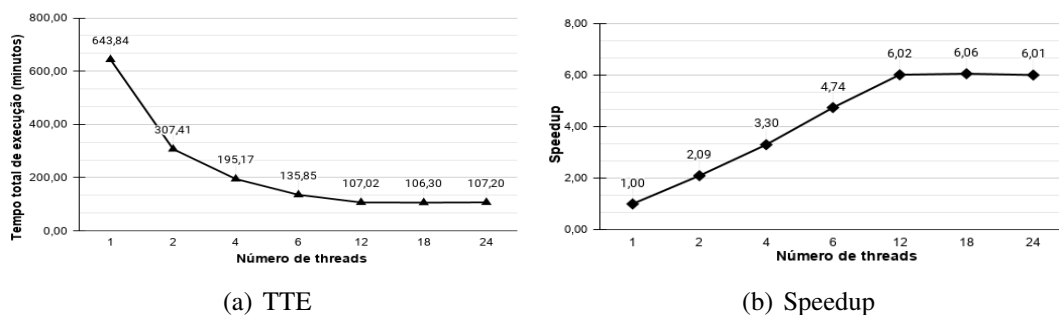


Figura 2. TTE e *Speedup*, variando o número de threads gerenciadas pelo Parsl

Para este experimento, o número máximo de threads do ThreadPoolExecutor estará variando de 1 a 24 e não será explorado o recurso multithread do software bowtie2. A Figura 2(a) apresenta o Tempo Total de Execução (TTE), em minutos, e a Figura 2(b) o speedup. É possível observar que para todos os experimentos houve diminuição no TTE e um aumento no desempenho. Porém, a partir de 12 threads há uma estabilização no ganho, devido ao tamanho da carga de entrada, sendo 12 threads para o Parsl, a melhor configuração no SDumont nesse caso abordado. Dessa forma, o TTE ficou em 107,02 minutos e um ganho de desempenho de 6,02.

3.2. Análise de desempenho da paralelização do Bowtie2

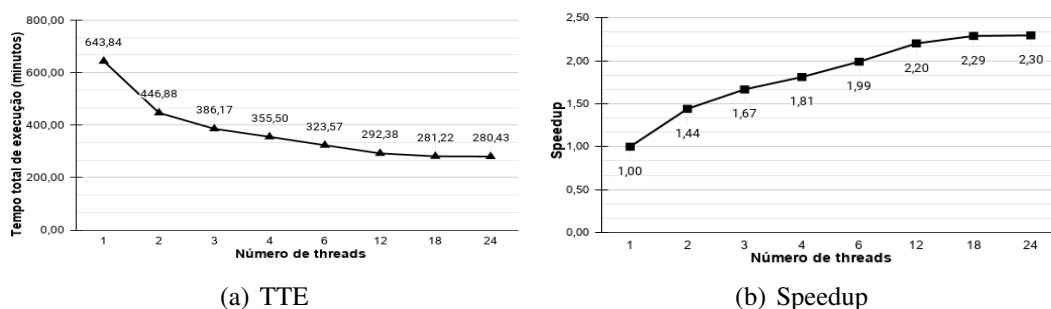


Figura 3. TTE e *Speedup*, variando o número de threads da aplicação Bowtie2

Para que seja explorado o recurso multithreading do software bowtie2, que compõe a primeira atividade, será fixado em 1 o número máximo de threads que o Parsl pode alocar,

⁵Disponível em <https://software.intel.com/content/www/us/en/develop/tools/vtune-profiler.html>. Último acesso: 14/08/2020.

forçando a execução serial. A Figura 3(a) apresenta o TTE em minutos e a Figura 3(b) o speedup, quando o bowtie2 varia o número de threads, de 1 a 24. É possível observar que para esse experimento, também, há uma estabilização no ganho de desempenho a partir de 12 threads, onde a melhor configuração ficou em 24 threads na aplicação bowtie2. Dessa forma o TTE ficou em 280,43 minutos e o ganho de desempenho foi 2,30. Esse experimento indica que há uma grande paralelização com a utilização do ThreadPoolExecutor, do Parsl, onde se obteve um melhor ganho de desempenho.

3.3. Análise de desempenho multithreading do Parsl e do Bowtie2

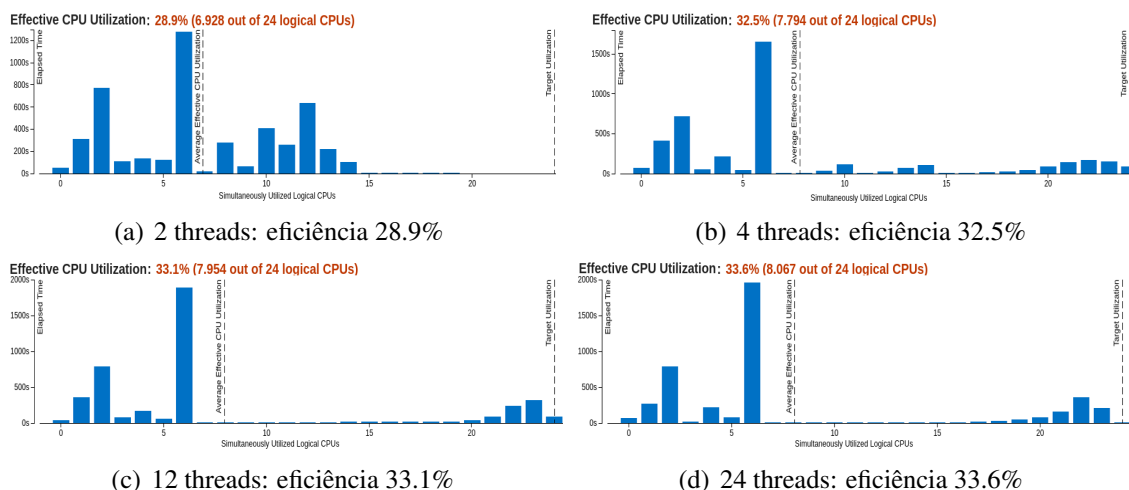


Figura 4. Eficiência do uso das CPUs disponíveis em um nó SDumont.

Para que sejam explorados todos os recursos do workflow, este experimento fixa o máximo de threads do ThreadPoolExecutor em 12, de acordo com a análise feita na subseção 3.1. E, as threads que variam são do software bowtie2. A Figura 4, apresenta uma análise do perfilador VTune onde é apresentado um histograma de eficiência pela utilização das CPUs. No histograma, é apresentado baseado no tempo total de execução, o tempo decorrido de cada núcleo que foram utilizados simultaneamente. O eixo vertical, é o tempo de execução em segundos e o eixo horizontal, é o número de núcleos disponíveis em um nó do SDumont. É possível observar que a partir de 4 threads, há o uso de todos os núcleos disponíveis, e a partir de 12 threads as eficiências são similares.

3.4. Análise variando o número de arquivos de entrada

Para esta seção, os experimentos foram realizados variando, de 4 a 10, a quantidades de arquivos de sequenciamento. Temos: para 4 arquivos, o total de 6.3GB; para 6 arquivos, foi a mesma formação de arquivos conforme as seções anteriores; para 8, foi feita a réplica dos arquivos com tamanho 1.5GB, totalizando 14GB; e para 10, foram replicados os arquivos com tamanhos 1.5GB, 1.6GB, 1.5GB e 1.7GB, com um total de 17.3GB. Foi utilizado, também, para esse experimento o modelo ThreadPoolExecutor com 12 threads e o bowtie2 com 1 thread, de acordo com a análise feita na subseção 3.1.

Na Figura 5, é apresentada uma análise do perfilador VTune, que indica a eficiência no uso das CPUs. O experimento indica que quanto maior o número de arquivos, maior é a eficiência. Dessa forma, o melhor formato, nesse cenário, é a execução com 10 arquivos.

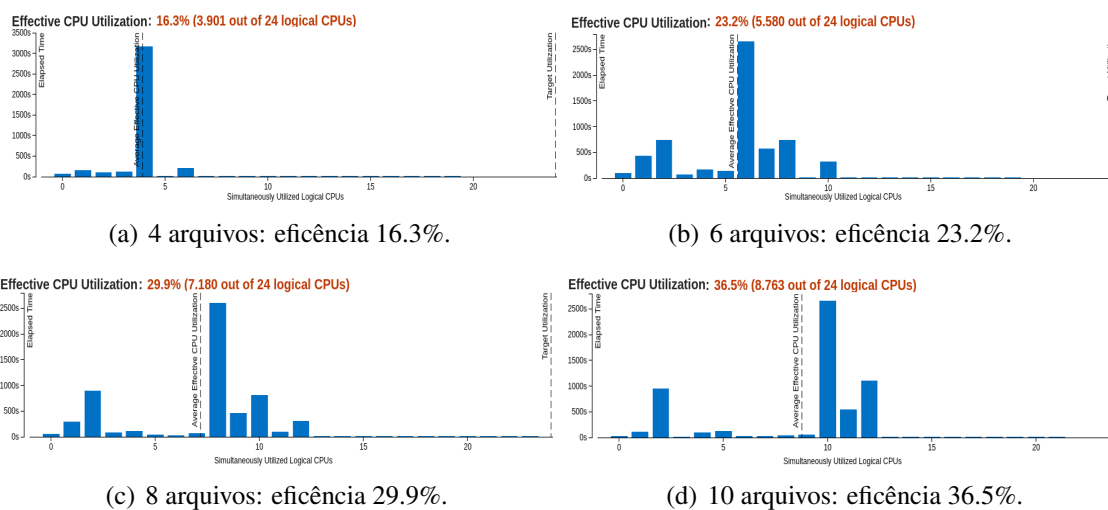


Figura 5. Eficiência No uso das CPUs disponíveis em um nó do SDumont.

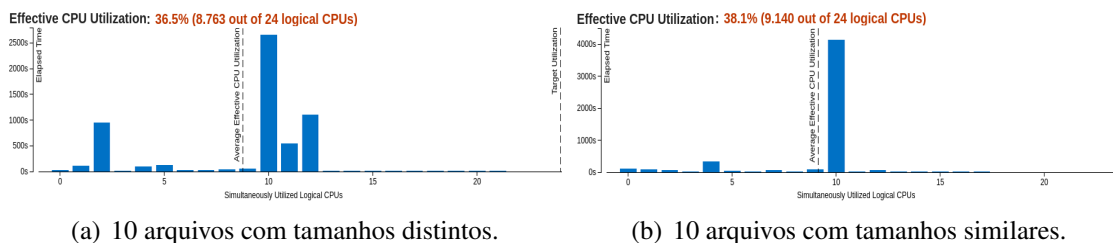


Figura 6. Eficiência de 10 arquivos 6(a) desbalanceados e 6(b) balanceados.

Com a análise realizada, foi levantada a hipótese de que cada arquivo é executado em um núcleo do processador, como pode ser observado na Figura 5. Pois, dado a quantidade n de arquivos, na maior parte do tempo de execução são utilizados n núcleos, simultaneamente. Também, é possível observar o desbalanceamento de carga, ou seja, o trabalho é distribuído de uma maneira irregular, de forma que os núcleos com menos tarefas aguardem o fim da execução dos demais. A distribuição é feita de acordo com a quantidade de arquivos, e esses arquivos, por terem tamanhos distintos, ocasionam o desbalanceamento. Na Figura 6(b) é apresentada uma análise do VTune, em um cenário sintético, onde são utilizados 10 arquivos com aproximadamente o mesmo tamanho, onde cada arquivo varia o tamanho de 1.5GB a 1.7GB. Com esse cenário é possível demonstrar o balanceamento de carga. Como todos os arquivos têm um tamanho similar, requerendo o mesmo tempo computacional, os demais núcleos não ficam ociosos.

4. Conclusão

O workflow ParslRNA-Seq foi modelado pelo Parsl e os experimentos de desempenho no SDumont apresentam resultados que confirmam a funcionalidade, facilitando a execução de atividades intensivas, complexas e computacionalmente custosas. A análise dos dados gerados foram validados pelo bioinformata através da comparação entre os dados obtidos por execução manual no SDumont e das execuções do workflow RNA-Seq no Galaxy, comprovando a redução no tempo de execução e as adaptabilidades em ambientes de

CAD. O código fonte do ParslRNA-Seq está disponível no GitHub⁶.

Cabe também ressaltar que a ferramenta Parsl ainda não está sendo utilizada em todo o seu potencial, pois, como levantado na seção 2.3, os experimentos foram realizados utilizando apenas um nó computacional, para fins de análises de desempenho. Como passos futuros, serão realizadas execuções em larga escala, explorando mais os recursos do Parsl, como HighThroughputExecutor e incrementando a complexidade de execução do ParslRNA-Seq. Além do uso de perfiladores para fins de otimização, também será estudado a melhor forma de manipular e armazenar os dados de proveniência. Este workflow também será acoplado ao Bioinfo-Portal⁷ [Ocaña et al. 2020], que é um portal, hospedado no LNCC, voltado ao fortalecimento das pesquisas envolvendo o uso da bioinformática.

5. Agradecimentos

Agradecemos ao Laboratório Nacional de Computação Científica (LNCC/MCTI, Brasil) por prover os recursos de CAD do supercomputador SDumont e pelo apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Referências

- Anders, S., Pyl, P. T., and Huber, W. (2014). HTSeq—a Python framework to work with high-throughput sequencing data. *Bioinformatics*, 31(2):166–169.
- Babuji, Y., Woodard, A., Li, Z., Katz, D. S., Clifford, B., Kumar, R., Lacinski, L., Chard, R., Wozniak, J., Foster, I., Wilde, M., and Chard, K. (2019). Parsl: Pervasive parallel programming in python. In *28th ACM International Symposium on High-Performance Parallel and Distributed Computing (HPDC)*.
- Braghetto, K. R. and Cordeiro, D. (2014). Introdução à modelagem e execução de workflows científicos. *Atualizações em Informática. Ied. Porto Alegre: SBC*, pages 1–40.
- Cordeiro, D., Braghetto, K. R., Goldman, A., and Kon, F. (2013). Da ciência à e-ciência: paradigmas da descoberta do conhecimento. *Revista USP*, (97):71–81.
- Langmead, B. and Salzberg, S. L. (2012). Fast gapped-read alignment with bowtie 2. *Nature methods*, 9(4):357.
- Love, M. I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15(12):550.
- Mattos, A., Silva, F., Ruberg, N., and Cruz, M. (2008). Gerência de workflows científicos: uma análise crítica no contexto da bioinformática. *COPPE/UFRJ*.
- Ocaña, K. A. C. S., Galheigo, M., Osthoff, C., Gadelha, L., Porto, F., Gomes, A., Oliveira, D., and Vasconcelos, A. T. (2020). Bioinfoportal: A scientific gateway for integrating bioinformatics applications on the brazilian national high-performance computing network. *Future Generation Computer Systems*, 107:192–214.
- Silva, R. R. and Yokoyama, R. S. (2011). Avaliação do desempenho de threads em user level utilizando sistema operacional linux. *Revista de Informática Teórica e Aplicada*.

⁶Disponível em <https://github.com/lucruzz/rna-seq>. Último acesso: 14/08/2020.

⁷Disponível em <https://bioinfo.lncc.br/>. Último acesso: 01/10/2020.