Avaliação do desempenho relativo de bancos de dados NoSQL para arquivos de genótipos

Vinícius Junqueira Schettino[1], **Arthur Lorenzi Almeida**[1], **Leojayme Rodrigues Manso Silva**[1], **Wagner Arbex**[1,2,*]

[1]Universidade Federal de Juiz de Fora – UFJF
Dep. de Ciência da Computação – DCC
Campus Universitário, 36.036-900, Juiz de Fora, MG, Brasil

[2]Empresa Brasileira da Pesquisa Agropecuária – Embrapa
R. Eugênio do Nascimento, 610, 36.038-330, Juiz de Fora, MG, Brasil

vinicius.schettino@ice.ufjf.br, lorenzi@ice.ufjf.br,
leojayme.manso@ice.ufjf.br, wagner.arbex@{ufjf.edu.br,embrapa.br}

Abstract. *The bioinformatics and the genomics work with nonstandard database. The classic databases are characterized by tabular design of data set and handling thereof by RDBMS. The genotype files are instances of non-classical databases and are characterized by being generated as text files, with unbalanced data, high dimensionality and the large volume of data, among other things. The RDBMS have not shown a great solution for the processing of such bases and, therefore, this study is evaluating the performance of NoSQL databases representing of two different data model families from testing scenarios for handling genotype files.*

Resumo. *A bioinformática e a genômica trabalham com bases de dados fora do padrão tradicional ou clássico que, por sua vez, caracterizam-se pela organização tabular e pelo tratamento destas em SGBDRs. Arquivos de genótipos são exemplos de bases de dados não clássicas e são caracterizados por serem gerados como arquivos textos, com dados desbalanceados, com alta dimensionalidade e por ocuparem muito espaço, entre outros aspectos. Os SGBDRs não têm se mostrado uma boa solução para o tratamento de tais bases e, portanto, o presente trabalho busca avaliar o desempenho relativo entre bancos de dados NoSQL que representam duas famílias de diferentes modelo de dados, a partir de cenários de teste para a manipulação de arquivos de genótipo.*

1. Introdução

Uma das atividades mais complexas das pesquisas em bioinformática tem sido a manipulação de banco de dados, que, em geral, tratam com 3 diferentes tipos de conjuntos de dados genômicos, quais sejam: os dados de sequenciamento ou sequências de nucleotídeos, também conhecidos como *reads*; os dados de genotipagem por marcadores moleculares do tipo SNP ou, simplesmente, "genótipo" ou "marcadores SNP"; e, ainda, os muitos arquivos de metadados dos *reads* e genótipos.

*Autor correspondente.

Os genótipos com marcadores moleculares do tipo SNP, segundo [Caetano 2009], surgiram na década passada, com as plataformas de genotipagem de marcadores moleculares de polimorfismo de base única ou *single polymorphism nucleotide* (SNP). No genoma humano, cerca de 0,1% dos nucleotídeos podem ser SNPs, segundo [Brookes 1999], o que pode determinar, aproximadamente, 3 milhões de SNPs por indivíduo, se for considerada a genotipagem de todos os SNPs do genoma do indivíduo.

As tecnologias que permitem a "leitura" de *chips* de DNA para a genotipagem de marcadores SNP, são divididas em baixa, média e alta densidade, possibilitando a genotipagem de alguns milhares de nucleotídeos – p. ex., 3 mil –, até centenas de milhares, que ultrapassam a 700 mil o número de marcadores SNP genotipados em um único ensaio.

A redução do custo dos serviços de genotipagem tem permitido aos projetos fazerem experimentos com *chips* de média ou alta densidade e, pela relação "custo $\times$ benefício", os *chips* de média densidade – que geram genótipos de algumas dezenas de milhares até mais de 100 mil marcadores SNP –, talvez sejam os mais utilizados.

Neste contexto, o presente trabalho tem como objetivo avaliar o desempenho relativo dos SGBDs MongoDB e Tarantool[1], que representam duas diferentes "famílias" de bancos de dados NoSQL, a partir da manipulação de arquivos com dados de genotipagem. Para tanto, foi desenvolvido e executado um *benchmark*, a partir da geração de um arquivo de genótipos simulados de 5 mil indivíduos, com 56 mil marcadores SNP para cada indivíduo.

2. Considerações sobre bancos de dados NoSQL e trabalhos relacionados

Os sistemas de bancos de dados NoSQL, ou "not-only SQL", surgiram para atender às necessidades da computação científica e de novos paradigmas, tais como, *big data* e *data science*, assim como, uma alternativa para os problemas de escalabilidade em armazenamento, paralelismo e gestão de grandes volumes de dados não estruturados, comuns na manipulação de bases de dados de genótipos.

[Edlich 2016] lista cerca de 12 modelos ou "famílias" de bancos de dados NoSQL, sendo os sistemas baseados "em colunas" (*column family database* ou *wide column store*), "em documentos" (*document store*), "em chave-valor" (*key-value* ou *tuple store*) e "em grafos" (*graph databases*) os mais utilizados ou referenciados, como pode ser visto em [Hecht and Jablonski 2011], [Li and Manoharan 2013], [Veronika Abramova; Jorge Bernardino and Pedro Furtado 2014] e [Aniceto et al. 2015].

Para este trabalho foram utilizados os bancos de dados MongoDB e Tarantool que representam, respectivamente, os SGBDs baseados em documentos e chave-valor. Além de serem utilizados em diversos trabalhos relacionados, como os citados anteriormente.

3. O experimento e seus materiais e métodos

O experimento baseou-se no desenvolvimento e na execução de um *benchmark* com o uso do Yahoo! Cloud Serving Benchmark (YCSB) 0.7, descrito em [Cooper et al. 2010], o qual é amplamente utilizado para a comparação e avaliação de desempenho de SGBDs. Seu ambiente de execução utilizou os SGBDs MongoDB 3.2.3 e Tarantool 1.6.8, em suas configurações padrões, com Debian 7.0 em um hardware de 16 GB de memória RAM.

[1]Informações sobre MongoDB e Tarantool em [MongoDB, Inc. 2016] e [Mail.Ru Group 2016].

O YCSB é composto de um gerador de dados e um conjunto de testes de desempenho para avaliar operações de leitura, inserção, atualização etc.. Cada um dos cenários de teste é associado a uma *workload*, ou "carga de trabalho", e é definido por um conjunto de características, como, p. ex., a porcentagem de operações de leitura/atualização, o número total de transações ou o número de registros da *workload*. O conjunto padrão de *workloads* do YCSB não atendeu às exigências do experimento a ser realizado, por não reproduzir as características de um arquivo de genótipo. Assim, o primeiro passo do experimento foi a criação de uma *workload* personalizada, simulando uma população de 5 mil indivíduos genotipados e, cada indivíduo, com um genótipo hipotético de 56 mil marcadores. O que gerou uma carga de 5 mil registros, com 56 mil campos por registro e 1 byte por cada campo.

O passo seguinte consistia na criação do banco de dados, no ambiente do YCSB, para cada um dos SGBDs utilizados e, o terceiro passo, na execução das operações dos cenários de teste em cada SGBD e na recuperação dos resultados para análise.

Houve, ainda, a necessidade de se definir as operações a serem executadas nos testes, pois, pela natureza dos trabalhos com arquivos de genótipo, operações de edição e inserção são pouco relevantes. Para tais arquivos, os desempenhos de leitura e/ou atualização aliados a escalabilidade são mais importantes e mais próximos de situações relevantes ao contexto dos mesmos. Portanto, no terceiro passo, o experimento foi dividido em duas etapas, os cenários C1, com as operações de carga e inserção, e C2, com as operações de leitura e atualização.

4. Resultados e análise do experimento

A Tabela 1 apresenta o resultado do *benchmark* entre os bancos de dados MongoDB e Tarantool, nas duas etapas do experimento.

Tabela 1. Resultado do *benchmark* entre os sistemas MongoDB e Tarantool.

	C1			C2		
	tempo de exec. ms	*throughput* op/s	des. relativo %	tempo de exec. ms	*throughput* op/s	des. relativo %
MongoDB	272.229,0	18,37	111,13	152.252,0	65,60	1,84
Tarantool	302.449,0	16,53	89,98	2.802,1	3.568,75	5440,17

Para C1 foram executadas 5 mil operações de inserção, para os 5 mil registros da *workload*, quando o MongoDB obteve um desempenho ligeiramente superior ao Tarantool, explicado pelo *throughput*. Como consequência, o tempo de execução do MongoDB foi pouco superior a 4,5 minutos e o do Tarantool pouco superior a 5 minutos.

A execução de C2 consistia em 10 mil operações divididas uniformemente entre operações de leitura e atualização. O resultado para C2 mostrou-se diferente, pois enquanto o tempo de execução do MongoDB foi pouco superior a 2,5 minutos, o do Tarantool, não chegou a 3 segundos, o que, também, pode ser explicado pelo *throughput*.

O desempenho relativo foi tomado com base no *throughput* dos cenários do experimento e, pela análise de C1, MongoDB mostrou-se mais eficiente e cerca de 11% mais rápido do que Tarantool, que, por sua vez, não chegou a 90% do resultado obtido pelo MongoDB. Todavia, mediante à C2, o resultado se inverte e o MongoDB tem um

desempenho inferior a 2%, se comparado ao desempenho do Tarantool que, por sua vez, apresentou um desempenho marcante, sendo mais de 5400 vezes mais rápido do que o MongoDB.

5. Conclusão

O tratamento adequado de arquivos de genótipos por uma solução de banco de dados é de grande importância para projetos em bionformática e genômica e, notadamente, os recursos clássicos supridos por SGBDRs não cumprem esta tarefa.

A análise dos resultados permite concluir que, para armazenamento, leitura e atualização de arquivos de genótipos, o uso de um banco de dados NoSQL, como o Tarantool, deve ser considerado, se comparado a sistemas com modelo de dados baseados em documentos, como é o caso do MongoDB.

A partir desta conclusão, devem ser apreciadas em investigações futuras (i) a hipótese de que bancos de dados NoSQL, com modelos de dados baseados em chave-valor, apresentam desempenho superior em relação a bancos de dados NoSQL com outros modelos de dados, para arquivos de genótipo; e (ii) a complementação da abordagem utilizada nos experimentos deste trabalho com cenários que avaliem escalabilidade em bancos de dados NoSQL.

Referências

Aniceto, R., Xavier, R., Guimarães, V., Hondo, F., Holanda, M., Walter, M. E., and Lifschitz, S. (2015). Evaluating the cassandra NoSQL database approach for genomic data persistency. *International Journal of Genomics*, 2015.

Brookes, A. J. (1999). The essence of SNPs. *Gene*, 2(234):177–186.

Caetano, A. R. (2009). Marcadores SNP: conceitos básicos, aplicações no manejo e no melhoramento animal e perspectivas para o futuro. *Rev. Bras. de Zootecnia*, 38:64–71.

Cooper, B. F., Silberstein, A., Tam, E., Ramakrishnan, R., and Sears, R. (2010). Benchmarking cloud serving systems with ycsb. In *Proceedings of the 1st ACM Symposium on Cloud Computing*, SoCC '10, pages 143–154, New York, NY, USA. ACM.

Edlich, S. (2016). NoSQL. http://www.nosql-database.org/.

Hecht, R. and Jablonski, S. (2011). NoSQL evaluation: A use case oriented survey. In *2011 International Conference on Cloud and Service Computing*, pages 336–341. IEEE.

Li, Y. and Manoharan, S. (2013). A performance comparison of sql and nosql databases. In *Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on*, pages 15–19.

Mail.Ru Group (2016). Tarantool. http://tarantool.org/.

MongoDB, Inc. (2016). MongoDB for giant ideas. https://www.mongodb.org/.

Veronika Abramova; Jorge Bernardino and Pedro Furtado (2014). Experimental Evaluation of Nosql Databases. *International Journal of Database Management Systems (IJDMS)*, 6(3):1–16.