Avaliação Experimental da Configuração do Threshold de Balanceamento de Réplicas no HDFS Balancer

Rhauani Weber Aita Fazul[1], **Patrícia Pitthan Barcelos**[2]

[1]Laboratório de Sistemas de Computação (LSC)
[2]Pós-Graduação em Ciência da Computação (PGCC)
Universidade Federal de Santa Maria (UFSM)
Santa Maria – RS – Brasil

{rwfazul, pitthan}@inf.ufsm.br

Resumo. *O HDFS Balancer opera através de um threshold que determina o nível de equilíbrio a ser atingindo com a redistribuição dos dados. Definir um threshold ideal, entretanto, apresenta-se com um desafio para o administrador do sistema. Este trabalho analisa o comportamento do HDFS através de experimentos com variações na configuração do threshold. Os resultados demostram as melhorias de desempenho impulsionadas pelo balanceamento do cluster.*

1. Introdução

A replicação de dados é fundamental para o funcionamento do Sistema de Arquivos Distribuído do Apache Hadoop[1], o HDFS. Para o armazenamento das réplicas nos `DataNodes` (DNs), o servidor mestre do HDFS (o `NameNode`) segue uma Política de Posicionamento de Réplicas (PPR), que fornece um bom equilíbrio entre confiabilidade e disponibilidade dos dados [White 2015]. De todo modo, nem sempre é possível impedir o desbalanceamento na distribuição das réplicas no *cluster*.

O `HDFS Balancer` [Shvachko et al. 2010] é uma solução disponibilizada pelo Hadoop que redistribui os dados já armazenados no sistema de arquivos visando o balanceamento de réplicas. Além de definir o momento adequado para a execução do `HDFS Balancer`, o administrador do sistema deve configurar o *threshold* de balanceamento. O *threshold* controla o funcionamento da ferramenta ao determinar quando o *cluster* pode ou não ser considerado balanceado. Escolher um bom valor para o *threshold* é essencial para otimizar a operação de balanceamento de réplicas. *Thresholds* menores permitem atingir um maior equilíbrio, porém demandam um maior esforço para a movimentação dos dados. Já *thresholds* maiores reduzem o tempo de execução e a largura de banda consumida pelo balanceador, entretanto resultam em uma distribuição menos equilibrada.

Este trabalho analisa a influência do *threshold* de balanceamento no funcionamento do `HDFS Balancer` e no desempenho do sistema de arquivos do Hadoop. Para tal, diferentes experimentos com variações no valor do *threshold* foram conduzidos. Com isso, permite-se investigar as possíveis otimizações impulsionadas pelo equilíbrio na distribuição das réplicas e evidenciar o custo demandado para tal operação.

O artigo está organizado em cinco seções. A Seção 2 descreve as principais causas e os problemas do desbalanceamento no HDFS. A Seção 3 apresenta o balanceador de réplicas nativo do Apache Hadoop. A Seção 4 exibe e discute os resultados obtidos. Por fim, a Seção 5 aponta as considerações finais e direciona os trabalhos futuros.

[1]`https://hadoop.apache.org/`

2. Desbalanceamento de Réplicas

O HDFS implementa uma estratégia de armazenamento que consiste na segmentação dos arquivos em blocos de dados de tamanho fixo, que são replicados com base em um Fator de Replicação (FR) e distribuídos através do *cluster*. Embora haja um esforço em manter um balanceamento mínimo no posicionamento das réplicas entre os DNs, nem sempre é possível impedir o desequilíbrio. Dentre as principais causas do desbalanceamento de réplicas no HDFS estão [Hortonworks 2019]: (i) a adição de novos DNs ao *cluster*; (ii) o comportamento da aplicação do cliente; e (iii) a alocação dos blocos satisfazendo a PPR.

O Hadoop explora a localidade dos dados ao mover as tarefas de computação para onde as réplicas estão armazenadas, evitando mover os dados em si e possibilitando que o acesso/processamento seja feito localmente [White 2015]. A medida que o desbalanceamento se intensifica – além de gerar sobrecarga para os DNs com maior utilização –, a localidade é afetada e a largura de banda do *cluster* passa a ser consumida para realizar transferência de dados, prejudicando o desempenho de aplicações focadas em entrada e saída (E/S) intensiva. Para mitigar esses problemas, o Hadoop disponibiliza uma solução voltada ao balanceamento de réplicas no HDFS, conforme apresentado na Seção 3.

3. HDFS Balancer

O `HDFS Balancer` [Shvachko et al. 2010] é uma ferramenta integrada na distribuição do Hadoop responsável pela análise do posicionamento dos blocos presentes no sistema de arquivos, cabendo a ele tomar as decisões referentes à redistribuição de dados entre os DNs. A partir de sua política de execução padrão, o `HDFS Balancer` opera iterativamente movimentando blocos de DNs que apresentarem uma alta utilização (origem) para DNs que possuírem um menor volume de dados armazenado (destino) [White 2015]. A execução da ferramenta é disparada sob demanda pelo administrador do *cluster*.

A operação do balanceador é guiada por um *threshold* (porcentagem no intervalo de 0% a 100%), que é passado como parâmetro para sua execução. Um mesmo DN pode possuir um ou mais dispositivos de armazenamento de diferentes tipos (e.g. disco rígido e SSD). Sendo $G_{i,t}$ o grupo de dispositivos do tipo t de um determinado DN, o *threshold* limita a diferença máxima que a utilização do $G_{i,t}$ ($U_{i,t}$) e a utilização média dos dispositivos do tipo t do *cluster* ($U_{\mu,t}$) pode assumir [Hortonworks 2019]. Quando a utilização de todos os grupos estiver dentro dos limites inferior e superior determinados com base no *threshold* (i.e., $U_{\mu,t} - \textit{threshold}$ e $U_{\mu,t} + \textit{threshold}$), o *cluster* é tido como balanceado.

Ao reduzir o *threshold* aumenta-se o nível de equilíbrio na distribuição das réplicas no *cluster*, todavia maior o esforço demandado, em termos de processamento e de transferência de dados, para efetuar o balanceamento. A seguir, na Seção 4, investiga-se o impacto do *threshold* na operação do balanceador e no funcionamento do HDFS.

4. Experimentos e Discussão

A experimentação foi realizada na plataforma GRID'5000[2] com o Hadoop (versão 2.9.2) operando em modo totalmente distribuído. O ambiente de testes consistiu em 10 nodos

[2]Grid'5000 é uma plataforma para experimentos apoiada por um grupo de interesses científicos hospedado pelo Inria e incluindo CNRS, RENATER e diversas Universidades, bem como outras organizações (mais detalhes em `https://www.grid5000.fr`).

(modelo *Dell PowerEdge R640*) configurados no *cluster* $gros$ do *site* $Nancy$, cada um com 1 processador Intel Xeon Gold 5220 (Cascade Lake-SP, 2.20GHz, 18 cores/CPU), 96GB de memória RAM, capacidade de armazenamento SDD (SATA) de 480GB e 2 conexões *Ethernet* de 25Gbps cada, executando uma distribuição Debian 10 (*buster*).

Para a carga dos dados utilizou-se o `TestDFSIO` [White 2015], um *benchmark* distribuído que testa o desempenho do HDFS com operações de E/S intensivas. Com o `TestDFSIO`, foram escritos 25 arquivos de 25GB cada e FR padrão de 3, totalizando um volume de dados de $1{,}85$TB. Após a escrita, o sistema de arquivos ficou com uma utilização média de $50{,}55\%$. De forma a avaliar possíveis melhorias de desempenho impulsionadas pelo equilíbrio de réplicas, avaliou-se o comportamento do HDFS com a distribuição dos dados baseada na PPR (i.e., sem balanceamento) e após a execução do `HDFS Balancer` considerando cinco cenários com configurações distintas do *threshold* de balanceamento, sendo eles: (i) 15%; (ii) $12{,}5\%$ (iii) 10%; (iv) $7{,}5\%$; e (v) 5%.

A Tabela 1 apresenta, para cada um dos DNs, a sua ocupação em GB (O_{GB}) e a porcentagem de utilização ($U_{\%}$) em cada um dos cenários de teste. Para o cenário sem balanceamento, esses valores equivalem ao estado do HDFS após a escrita dos arquivos e a distribuição dos blocos baseada na PPR. Para os demais cenários, os valores representam o estado de cada um dos DNs do *cluster* após a execução do `HDFS Balancer` com a respectiva configuração de *threshold* (*Th*). Observa-se como, em cada um dos cenários com balanceamento de réplicas, a utilização final dos DNs passa a respeitar os limites inferior ($U_{\mu,SSD} - threshold$, i.e., $50{,}55\% - Th$) e superior ($U_{\mu,SSD} + threshold$, i.e., $50{,}55\% + Th$) considerados pelo balanceador.

Tabela 1. Ocupação (O_{GB}) e utilização ($U_{\%}$) dos nodos em cada cenário de teste.

DN	sem bal.		*Th* = 15%		*Th* = 12,5%		*Th* = 10%		*Th* = 7,5%		*Th* = 5%	
	O_{GB}	$U_{\%}$	O_{GB}	$U_{\%}$	O_{GB}	$U_{\%}$	O_{GB}	$U_{\%}$	O_{GB}	$U_{\%}$	O_{GB}	$U_{\%}$
DN_{01}	138,93	35,17	144,12	36,48	160,00	40,50	161,62	40,91	170,42	43,14	180,03	45,57
DN_{02}	142,34	36,03	152,19	38,52	197,54	50,00	163,02	41,26	189,60	47,99	195,78	49,56
DN_{03}	144,19	36,50	162,26	41,07	237,60	60,14	185,70	47,00	171,04	43,29	195,78	49,56
DN_{04}	191,16	48,39	191,58	48,49	164,53	41,65	213,16	53,95	205,81	52,09	182,30	46,14
DN_{05}	140,33	35,52	244,53	61,90	235,33	59,57	173,78	43,99	200,66	50,79	180,25	45,62
DN_{06}	145,64	36,86	239,99	60,75	154,45	39,09	224,14	56,73	213,79	54,11	200,06	50,64
DN_{07}	259,40	65,66	142,74	36,13	185,61	46,98	222,48	56,31	173,56	43,93	203,48	51,50
DN_{08}	286,38	72,49	220,34	55,77	166,67	42,19	160,38	40,60	213,16	53,95	184,82	46,78
DN_{09}	293,07	74,18	149,94	37,95	235,46	59,60	219,33	55,52	172,42	43,64	190,34	48,18
DN_{10}	150,63	38,13	246,42	62,37	158,11	40,02	169,32	42,86	185,12	46,86	180,16	45,60

Para a análise de desempenho no HDFS, realizou-se, em cada cenário de teste, 15 execuções distintas do *benchmark* `TestDFSIO` destinadas à leitura total dos dados armazenados no sistema. A Tabela 2 exibe as médias aritméticas dos valores de tempo de execução, *throughput* de leitura e taxa de E/S alcançados nas 15 execuções. Adicionalmente, para cada uma dessas três métricas, exibe-se a variação percentual dos cenários com balanceamento em relação ao cenário sem balanceamento. A variação percentual é dada pela equação $((T_b - T_a)\ /\ T_a \times 100)$, onde T_a e T_b equivalem, respectivamente, às médias da métrica em análise no cenário sem balanceamento e com o uso do balanceador com o respectivo *threshold* (*Th*). Quando negativa, a variação obtida representa uma redução. Dessa forma, percebe-se uma relação entre o desempenho do HDFS e o

equilíbrio das réplicas armazenadas no sistema (quanto menor o *threshold*, maior o nível de balanceamento atingido). Assim, com a localidade dos dados podendo ser melhor explorada, é possível aprimorar o funcionamento do HDFS ao reduzir o tempo necessário para a leitura dos dados e aumentar o *throughput* e taxa média de E/S do sistema.

Tabela 2. Comportamento do HDFS em cada cenário de teste.

Métrica	sem bal.	*Th* = 15%	*Th* = 12,5%	*Th* = 10%	*Th* = 7,5%	*Th* = 5%
Tempo de execução (s)	258,85	233,51	223,51	212,69	207,94	192,09
Variação percentual (%)	-	-9,79	-13,65	-17,83	-19,67	-25,79
Throughput médio (MB/s)	152,62	164,42	169,74	177,14	182,04	190,35
Variação percentual (%)	-	7,73	11,22	16,07	19,28	24,72
Taxa média de E/S (MB/s)	161,79	172,80	178,56	186,41	189,95	201,24
Variação percentual (%)	-	6,81	10,37	15,22	17,41	24,38
Tempo de balanceamento (s)	-	4015,76	4933,29	5967,74	6234,57	9867,83
Dados movimentados (GB)	-	52,75	96,38	114,38	142,00	211,88

Ao final da Tabela 2 exibe-se o tempo de execução do `HDFS Balancer` (em segundos) e o volume de dados transferidos entre os DNs durante a operação de balanceamento. Quanto menor o *threshold* maior o esforço necessário (em tempo de execução e largura de banda consumida) para equilibrar a distribuição das réplicas no *cluster*. Sendo assim, cabe ao administrador do sistema avaliar o *trade-off* entre as melhorias desempenho impulsionadas pelo balanceamento de réplicas no HDFS e custo para sua operação.

5. Considerações Finais

Este trabalho analisou a influência do balanceamento de réplicas no funcionamento do HDFS. Os resultados obtidos demostraram que a execução do `HDFS Balancer` com valores de *threshold* menores permite maiores otimizações no desempenho do sistema, porém demandando maior tempo para o balanceamento e realizando um maior número de transferências de dados entre os nodos do *cluster*. Espera-se que, ao evidenciar o *trade-off* entre o desempenho impulsionado e o custo da operação do `HDFS Balancer` nos experimentos idealizados, seja possível auxiliar a tomada de decisão de administradores de *clusters* HDFS para a configuração ideal do *threhsold* de balanceamento.

Trabalhos futuros envolvem novos experimentos com variações na configuração do balanceador nativo do HDFS, tais como a largura de banda máxima destinada ao balanceamento em cada nodo, o número máximo de transferências concorrentes e a quantidade de *threads* para realizar a redistribuição das réplicas. Adicionalmente, pretende-se avaliar a aplicabilidade do `HDFS Balancer` com as configurações recomendadas apresentadas em [Hortonworks 2019] para os modos de execução *background* e *fast*.

Referências

Hortonworks (2019). "Balancing data across an HDFS cluster". `https://docs.cloudera.com/HDPDocuments/HDP3/HDP-3.1.4/data-storage/content/balancing_data_across_hdfs_cluster.html`. Dezembro.

Shvachko, K., Kuang, H., Radia, S., and Chansler, R. (2010). The hadoop distributed file system. In *Symposium on Mass Storage Systems and Technologies*, pages 1–10. IEEE.

White, T. (2015). *Hadoop: The Definitive Guide*. O'Reilly Media, Inc., 4 edition.