Eduardo Camilo Inacio

CARACTERIZAÇÃO E MODELAGEM MULTIVARIADA DO DESEMPENHO DE SISTEMAS DE ARQUIVOS PARALELOS

Dissertação submetida ao Programa de Pós-Graduação em Ciências da Computação para a obtenção do Grau de Mestre em Ciências da Computação. Orientador: Prof. Ph.D. Mario Antonio Ribeiro Dantas

Florianópolis

2015

Ficha de identificação da obra elaborada pelo autor através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Inacio, Eduardo Camilo
Caracterização e modelagem multivariada do desempenho de sistemas de arquivos paralelos / Eduardo Camilo Inacio ; orientador Mario Antonio Ribeiro Dantas - Florianópolis, SC, 2015.
96p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação.

Inclui referências

1. Ciência da Computação. 2. Modelo de desempenho. 3. Sistema de arquivos paralelo. 4. Caracterização de carga de trabalho. 5. Análise multivariada. I. Dantas, Mario Antonio Ribeiro. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Ciência da Computação. III. Título.

Eduardo Camilo Inacio

CARACTERIZAÇÃO E MODELAGEM MULTIVARIADA DO DESEMPENHO DE SISTEMAS DE ARQUIVOS PARALELOS

Esta Dissertação foi julgada aprovada para a obtenção do Título de "Mestre em Ciências da Computação", e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciências da Computação.

Florianópolis, 03 de fevereiro 2015.

Prof. Dr. Ronaldo dos Santos Mello
Coordenador

Banca Examinadora:

Prof. Ph.D. Mario Antonio Ribeiro Dantas
Orientador

Prof. Dr. Lisandro Zambenedetti Granville
Universidade Federal do Rio Grande do Sul

Profa. Dra. Patricia Della Méa Plentz
Universidade Federal de Santa Catarina

Prof. Dr. Ronaldo dos Santos Mello
Universidade Federal de Santa Catarina

Dedico esse trabalho à minha amada esposa, Alessandra, por seu apoio, incentivo e tolerância.

AGRADECIMENTOS

Gostaria de deixar registrado os meus sinceros agradecimentos ao meu orientador e amigo, Prof. Ph.D. Mario Antonio Ribeiro Dantas, por ter compartilhado comigo esse desafio. Agredeço por ter tido sua orientação não apenas no direcionamento deste trabalho, mas também na minha formação como pesquisador de maneira geral. Saiba que seu apoio, dedicação e sabedoria foram fundamentais durante toda a duração dessa pesquisa.

Gostaria de agradecer também ao meu amigo, Prof. Dr. Douglas Dyllon Jeronimo de Macedo, da Universidade Federal de Sergipe (UFS), pelo auxílio na definição do problema inicial dessa pesquisa e pelas inúmeras sugestões e críticas realizadas. Sua participação ativa nesse projeto foi muito apreciada.

Meus agradecimentos vão também para a pesquisadora e amiga, Francieli Zanon Boito, da Universidade Federal do Rio Grande do Sul (UFRGS). Seu apoio na realização dos experimentos e nossas conversas sobre sistemas de arquivos paralelos contribuíram muito para enriquecer esse trabalho.

Aos membros do Laboratório de Pesquisa em Sistemas Distribuídos (LAPESD) da Universidade Federal de Santa Catarina (UFSC), gostaria também de deixar os meus agradecimentos. As sugestões e críticas recebidas durante os seminários realizados no laboratório durante as várias etapas dessa pesquisa, contribuíram significativamente para o amadurecimento desse trabalho.

Gostaria também de agradecer a UFSC e ao Programa de Pós-Graduação em Ciências da Computação (PPGCC), representado por todos os seus professores, por compartilharem seus conhecimentos e experiências conosco. Ainda gostaria de agradecer a secretaria do PPGCC, representada pela Katiana, por seu apoio administrativo durante esse período.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela concessão de bolsa durante todo o período dessa pesquisa.

Ao grupo de interesse científico francês representado pelo Inria e incluindo CNRS, RENATER e diversas universidades e organizações, pelo uso dos recursos do Grid'5000 na condução dos experimentos desse trabalho de pesquisa.

À Microsoft Research, pela concessão de recursos na sua infraestrutura de nuvem pública Microsoft Azure, para a realização de experimentos.

This man, on one hand, believes that he
knows something, while not knowing [anything].
On the other hand, I - equally ignorant -
do not believe [that I know anything].

(Sócrates, 399 AC)

RESUMO

A quantidade de dados digitais gerados diariamente vem aumentando de forma significativa. Por consequência, as aplicações precisam manipular volumes de dados cada vez maiores, dos mais variados formatos e origens, em alta velocidade, sendo essa problemática denominada como *Big Data*. Uma vez que os dispositivos de armazenamento não acompanharam a evolução de desempenho observada em processadores e memórias principais, esses acabam se tornando os gargalos dessas aplicações. Sistemas de arquivos paralelos são soluções de software que vêm sendo amplamente adotados para mitigar as limitações de entrada e saída (E/S) encontradas nas plataformas computacionais atuais. Contudo, a utilização eficiente dessas soluções de armazenamento depende da compreensão do seu comportamento diante de diferentes condições de uso. Essa é uma tarefa particularmente desafiadora, em função do caráter multivariado do problema, ou seja, do fato de o desempenho geral do sistema depender do relacionamento e da influência de um grande conjunto de variáveis. Nesta dissertação se propõe um modelo analítico multivariado para representar o comportamento do desempenho do armazenamento em sistemas de arquivos paralelos para diferentes configurações e cargas de trabalho. Um extenso conjunto de experimentos, executados em quatro ambientes computacionais reais, foi realizado com o intuito de identificar um número significativo de variáveis relevantes, caracterizar a influência dessas variáveis no desempenho geral do sistema e construir e avaliar o modelo proposto. Como resultado do esforço de caracterização, o efeito de três fatores, não explorados em trabalhos anteriores, é apresentado. Os resultados da avaliação realizada, comparando o comportamento e valores estimados pelo modelo com o comportamento e valores medidos nos ambientes reais para diferentes cenários de uso, demonstraram que o modelo proposto obteve sucesso na representação do desempenho do sistema. Apesar de alguns desvios terem sido encontrados nos valores estimados pelo modelo, considerando o número significativamente maior de cenários de uso avaliados nessa pesquisa em comparação com propostas anteriores encontradas na literatura, a acurácia das predições foi considerada aceitável.

Palavras-chave: Modelo de desempenho. Sistema de arquivos paralelo. Caracterização de carga de trabalho. Análise multivariada. Big Data.

ABSTRACT

The amount of digital data generated dialy has increased significantly. Consequently, applications need to handle increasing volumes of data, in a variety of formats and sources, with high velocity, namely Big Data problem. Since storage devices did not follow the performance evolution observed in processors and main memories, they become the bottleneck of these applications. Parallel file systems are software solutions that have been widely adopted to mitigate input and output (I/O) limitations found in current computing platforms. However, the efficient utilization of these storage solutions depends on the understanding of their behavior in different conditions of use. This is a particularly challenging task, because of the multivariate nature of the problem, namely the fact that the overall performance of the system depends on the relationship and the influence of a large set of variables. This dissertation proposes an analytical multivariate model to represent storage performance behavior in parallel file systems for different configurations and workloads. An extensive set of experiments, executed in four real computing environments, was conducted in order to identify a significant number of relevant variables, to determine the influence of these variables on overall system performance, and to build and evaluate the proposed model. As a result of the characterization effort, the effect of three factors, not explored in previous works, is presented. Results of the model evaluation, comparing the behavior and values estimated by the model with behavior and values measured in real environments for different usage scenarios, showed that the proposed model was successful in system performance representation. Although some deviations were found in the values estimated by the model, considering the significantly higher number of usage scenarios evaluated in this research work compared to previous proposals found in the literature, the accuracy of prediction was considered acceptable.

Keywords: Performance model. Parallel file system. Workload characterization. Multivariate analysis. Big Data.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

A quantidade de dados digitais gerados diariamente vem aumentando de forma significativa. Segundo um estudo realizado pela IDC (2014), uma empresa de consultoria em tecnologia reconhecida mundialmente, espera-se que entre os anos de 2013 e 2020, a quantidade de dados digitais no mundo passe de 4,4 para 44 zettabytes. Um aumento de dez vezes em apenas sete anos. Essa quantidade exorbitante de dados remete diretamente à problemática de pesquisa aqui apresentada. Nesse contexto, observa-se a necessidade de uma melhor gestão e otimização no armazenamento desses dados. Esse enorme volume de dados tem origem nos mais variados dispositivos e aplicações.

Experimentos realizados no Grande Colisor de Hádrons (LHC, na sigla em inglês) do laboratório de física de partículas da Organização Europeia para Pesquisa Nuclear geram aproximadamente 30 petabytes de dados por ano (CERN, 2014). Outro exemplo de dispositivo é o radiotelescópio ASKAP (*Australian Square Kilometre Array Pathfinder*) que gera anualmente 75 petabytes de dados digitais (CSIRO, 2014). Na Internet, a rede social Facebook, por exemplo, processa diariamente mais de 500 terabytes de dados (GUPTA; GUPTA; SINGHAL, 2014). Em outras áreas, como redes de telemedicina e telessaúde e prospecção de petróleo, é possível encontrar sistemas de armazenamento com dezenas de terabytes de dados (MACEDO et al., 2009; SOARES et al., 2012a; HALLIBURTON, 2014). Esses sistemas têm em comum a necessidade de processar enormes *volumes* de dados, com os mais *variados* formatos e origens, no menor tempo possível, ou seja, com alta *velocidade*. A esse conjunto de problemas, caracterizados pelos três "Vs"(LANEY, 2001), dá-se o nome de *Big Data* (BERMAN, 2013).

Visando atender essa demanda crescente, o projeto de sistemas e arquiteturas de armazenamento precisa considerar os mais diversos fatores, tanto em nível de hardware quanto de software. Atualmente, enfrenta-se uma limitação física de desempenho dos dispositivos de armazenamento, cuja evolução não acompanhou o crescimento exponencial observado nos processadores (JANUKOWICZ; EASTWOOD, 2012). Uma alternativa que vem sendo adotada para mitigar essa limitação é o uso de sistemas de arquivos paralelos (SHAN; ANTYPAS; SHALF, 2008; SAINI et al., 2012; SOARES et al., 2012b).

Um sistema de arquivos paralelo é um tipo de sistema de arquivos distribuído, caracterizado pelo particionamento e distribuição dos arquivos por um aglomerado (*cluster*) de recursos computacionais com

papel de armazenamento (TANENBAUM; STEEN, 2006). Esse processo consiste basicamente da divisão do arquivo em faixas (*stripes*) de tamanhos iguais e da distribuição dessas faixas entre os servidores de dados disponíveis. Dessa forma, operações de leitura e escrita podem ser realizadas de forma paralela, aumentando a vazão de entrada e saída (E/S) do sistema. A agregação de recursos computacionais possibilita oferecer, sob um único e consistente espaço de nomes, um amplo repositório de armazenamento. Apesar das facilidades oferecidas, o uso eficiente de sistemas de arquivos paralelos depende da compreensão do relacionamento entre um grande número de variáveis, como a quantidade de nodos de armazenamento, tamanho da faixa, taxa de transmissão da rede de interconexão, entre outros.

Conforme indicado em um estudo realizado durante essa pesquisa, o processo de caracterização de carga de trabalho vem sendo amplamente adotado para melhoria e modelagem de desempenho em ambientes de computação de alto desempenho (CAD), nuvens computacionais e *Big Data* (INACIO; DANTAS, 2014). Esse processo permite descrever e reproduzir o comportamento de sistemas computacionais (ELNAFFAR; MARTIN, 2006).

Na literatura, são encontradas algumas propostas de modelagem do comportamento de sistemas de arquivos paralelos (SONG et al., 2011; NGUYEN; APON, 2012). Se observa nesses trabalhos um compromisso entre granularidade e acurácia. Enquanto modelos analíticos oferecem apenas aproximações com menor custo de execução (OBAIDAT; BOUDRIGA, 2010), propostas baseadas em simulação ou que utilizam informações de desempenho reais para fazer suas predições apresentaram acurácia de até 96%. Também foi observado que variáveis com efeito importante no desempenho dos sistemas de arquivos paralelos, como o tamanho da faixa e da cache de escrita, por exemplo, são pouco explorados na construção dos modelos.

A obtenção de uma caracterização e modelagem analítica mais generalizada do desempenho de sistemas de arquivos paralelos se faz importante para prover um melhor entendimento sobre o comportamento desses sistemas em diferentes cenários. O grande desafio na representação desse comportamento está na identificação do relacionamento entre as múltiplas variáveis envolvidas e o desempenho do sistema, ou seja, um problema de análise multivariada (EVERITT; HOTHORN, 2011). Esse conhecimento possibilitará que projetos de sistemas de armazenamento sejam mais eficientes com relação a utilização de recursos e possam extrair maior desempenho das configurações disponíveis.

1.1 OBJETIVO GERAL

Caracterizar e modelar o desempenho do armazenamento em sistemas de arquivos paralelos, visando prover um melhor entendimento dos fatores que influenciam o comportamento desses sistemas, diante de diferentes configurações e cargas de trabalho.

1.2 OBJETIVOS ESPECÍFICOS

O atendimento do objetivo geral desta pesquisa engloba o atendimento dos seguintes objetivos específicos:

- Identificar os fatores que influenciam na vazão e no tempo de resposta do armazenamento em sistemas de arquivos paralelos;
- Caracterizar o comportamento da escrita em sistemas de arquivos paralelos, considerando diferentes fatores e cenários de uso;
- Propor um modelo analítico multivariado para a predição da vazão e do tempo de resposta em sistemas de arquivos paralelos;
- Avaliar o modelo por meio de experimentos em ambientes reais de sistemas de arquivos paralelos.

1.3 MÉTODO

Para atender aos objetivos dessa pesquisa, o seguinte conjunto de etapas é realizado:

- Buscar na literatura abordagens e propostas de modelos para representação do comportamento do desempenho de sistemas de arquivos paralelos;
- Realizar experimentos com um sistema de arquivos paralelo utilizando diferentes plataformas computacionais, configurações e cargas de trabalho;
- Analisar o resultado dos experimentos, verificando o efeito de diversos fatores no desempenho geral do sistema;
- Construir um modelo analítico multivariado para o desempenho do armazenamento em sistemas de arquivos paralelos;

- Avaliar a acurácia do modelo com relação a medições de desempenho obtidas em ambientes reais.

1.4 ESCOPO

Esta dissertação tem como escopo o comportamento do desempenho da operação de escrita em sistemas de arquivos paralelos. Outras operações, como a operação de leitura, por exemplo, apesar de relevantes, não fazem parte do escopo deste trabalho. Da mesma forma, aspectos de segurança e dependabilidade também não são considerados. Esta decisão foi motivada por alguns aspectos. O primeiro deles refere-se a problemas identificados em trabalhos anteriores do grupo (SOARES, 2012; MACEDO, 2014) relacionados especificamente ao processo de armazenamento em sistemas de arquivos paralelos. Nos resultados experimentais destes trabalhos, verificou-se uma melhoria nas operações de leitura, porém um desempenho abaixo do esperado foi observado para as operações de escrita. Esta pesquisa teve início com foco específico na caracterização deste problema, visando considerar a operação de leitura em um segundo momento. Contudo, no decorrer da pesquisa, verificou-se a complexidade envolvida na caracterização e modelagem do desempenho do armazenamento em sistemas de arquivos paralelos. Visando obter resultados mais significativos e abrangentes, optou-se por restringir o escopo desta dissertação para a operação de escrita, postergando o estudo da operação de leitura para trabalhos futuros. Vale ressaltar que o desempenho da operação de escrita tem papel fundamental em diferentes tipos de aplicações (SATO et al., 2012; MACEDO et al., 2009; SOARES et al., 2012a), o que justifica o escopo deste trabalho.

1.5 ORGANIZAÇÃO DO TRABALHO

Essa dissertação está assim organizada. No Capítulo 2 são apresentados fundamentos básicos sobre sistemas de arquivos, sistemas de arquivos paralelos e caracterização e modelagem de desempenho. Uma discussão sobre propostas de modelagem do comportamento de sistemas de arquivos paralelos encontradas na literatura é apresentada no Capítulo 3. NO Capítulo 4 é descrita a proposta desta pesquisa, detalhando a formalização do modelo analítico multivariado construído. Os resultados da caracterização e da avaliação do modelo proposto são

apresentados no Capítulo 5. Considerações finais desta pesquisa e direcionamentos dos trabalhos futuros são apresentados no Capítulo 6. Por fim, no Apêndice A são apresentadas as publicações decorrentes desse estudo e no Apêndice B são apresentados resultados adicionais da avaliação do modelo proposto.

2 SISTEMA DE ARQUIVOS E MODELAGEM DE DESEMPENHO

Toda aplicação computacional precisa armazenar e recuperar informação (TANENBAUM, 2007). Fitas magnéticas, discos rígidos (HDD, do inglês *Hard Disk Drive*), discos ópticos e, mais recentemente, dispositivos de estado sólido (SSD, do inglês *Solid-State Drive*) vêm sendo utilizados para o armazenamento permanente de dados digitais. Para que as aplicações possam acessar essas informações de maneira eficiente e conveniente, os sistemas operacionais apresentam um mecanismo que abstrai as diferentes características de cada dispositivo de armazenamento: o *sistema de arquivos* (SILBERSCHATZ; GALVIN; GAGNE, 2008).

Uma visão geral sobre sistemas de arquivos é apresentada na Seção 2.1. Na Seção 2.2, discute-se sobre os sistemas de arquivos distribuídos, com ênfase nos sistemas de arquivos paralelos, solução tratada nessa pesquisa. Os principais aspectos relacionados a técnicas para caracterização e modelagem do desempenho de sistemas computacionais são abordados na Seção 2.3. A Seção 2.4 conclui esse capítulo com as considerações finais.

2.1 VISÃO GERAL

Um sistema de arquivos consiste basicamente de duas partes: uma coleção de *arquivos* e uma *estrutura de diretórios* (SILBERSCHATZ; GALVIN; GAGNE, 2008). O arquivo é a unidade lógica de informação no dispositivo de armazenamento secundário (TANENBAUM, 2007) e representa uma coleção nomeada de dados relacionados (SILBERSCHATZ; GALVIN; GAGNE, 2008). Além do nome e dos dados, o sistema operacional associa a cada arquivo um conjunto de informações, ou *metadados* (TANENBAUM, 2007). Exemplos destes metadados são o tipo do arquivo, tamanho, localização, proprietário, permissões e datas e horários de criação e atualização. Estes metadados são mantidos pelo sistema operacional na estrutura de diretórios.

A estrutura de diretórios tem também como função organizar os arquivos no sistema. A complexidade dessa organização depende da implementação do sistema de arquivos. A estrutura de diretórios pode ter desde um único nível até um complexo sistema hierárquico, composto por uma árvore de diretórios, sub-diretórios e arquivos (SILBERSCHATZ; GALVIN; GAGNE, 2008). Vale observar que em alguns sistemas opera-

cionais os diretórios, também chamados de *pastas*, são implementados como arquivos especiais (TANENBAUM, 2007).

2.1.1 Operações Básicas

Os sistemas operacionais oferecem um amplo conjunto de operações sobre arquivos e sobre a estrutura de diretórios. Em termos de arquivos, usualmente seis operações básicas são encontradas: criação, escrita, leitura, remoção, reposicionamento e truncagem (SILBERSCHATZ; GALVIN; GAGNE, 2008). A operação de *criação* consiste da alocação de espaço e do registro do arquivo na estrutura de diretórios. As operações de *escrita* e *leitura* proveem, respectivamente, o armazenamento e a recuperação de dados de uma posição do arquivo. A *remoção* de arquivos compreende a liberação do espaço utilizado pelo arquivo e a remoção do registro na estrutura de diretórios. O *reposicionamento* possibilita alterar a posição do ponteiro de leitura ou escrita dentro do arquivo. Por fim, a operação de *truncagem* consiste da remoção do conteúdo de um arquivo sem alteração dos seus atributos. Estas operações podem ser combinadas para prover outras mais complexas, como a operação de cópia de arquivos, por exemplo.

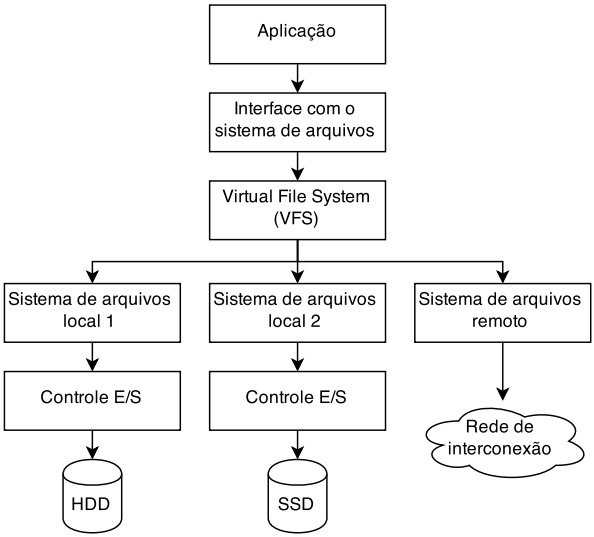
Com relação à estrutura de diretório, as operações normalmente encontradas são: criação, remoção, pesquisa, listagem, renomeação e varredura (SILBERSCHATZ; GALVIN; GAGNE, 2008). As operações de *criação* e *remoção*, respectivamente, criam ou removem entradas de arquivos no registro da estrutura de diretórios. Essa entradas recebem diferentes nomes dependendo do sistema operacional. Em sistemas UNIX e Linux essas entradas são chamadas de *file descriptors*, enquanto sistemas Windows se referem a essas entradas como *file handles* (SILBERSCHATZ; GALVIN; GAGNE, 2008). A operação de *pesquisa* permite localizar a entrada de um determinado arquivo. A *listagem* de um diretório retorna uma lista com os arquivos e diretórios contidos nesse. A operação de *renomeação* possibilita que o nome de um arquivo seja alterado. Por fim, a operação de *varredura* permite navegar por toda a estrutura de diretórios.

2.1.2 Estrutura de Armazenamento

Sistemas de arquivos são geralmente implementados em várias camadas, nas quais os serviços oferecidos pelas camadas inferiores são

utilizados para prover serviços mais elaborados para as camadas superiores (SILBERSCHATZ; GALVIN; GAGNE, 2008). A Figura 1 apresenta uma visão geral de uma estrutura de sistema de arquivos nos sistemas UNIX e Linux. Apesar de a discussão dessa seção poder ser estendida para sistemas Windows, nos concentramos nos sistemas UNIX e Linux. Essa escolha se deve ao fato da presença dominante destes sistemas entre os 500 ambientes computacionais de maior desempenho no mundo, segundo a lista de novembro de 2014 do TOP500 (2014).

Figura 1: Visão geral de uma estrutura de sistema de arquivos nos sistemas UNIX e Linux.



Fonte: Baseado em Silberschatz, Galvin e Gagne (2008).

A *aplicação* envia requisições para o sistema de arquivos por meio da *interface com o sistema de arquivos*. Essa interface é baseada em chamadas de sistema e entradas de arquivos (*file descriptors*). Nos sistemas UNIX e Linux, a *Portable Operating System Interface* (POSIX) (IEEE, 2004) é a interface padrão.

O padrão POSIX estabelece um modelo de entrada e saída (E/S) que enfatiza a consistência, de forma que todos os processos tenham uma visão atualizada dos dados (KIMPE; ROSS, 2014). Outros modelos de E/S, como o MPI-IO (MPI FORUM, 1997) e o HDF5 (THE HDF GROUP, 1997), são alternativas com consistência relaxada, visando prover maior desempenho para aplicações paralelas e com grande volume

e complexidade de dados.

O *Virtual File System* (VFS) oferece uma abstração para que as aplicações acessem diferentes sistemas de arquivos, sob uma mesma estrutura de diretórios, de forma transparente (KLEIMAN, 1986). Sua função consiste basicamente em identificar a qual sistema de arquivos pertence um determinado arquivo e ativar as operações específicas do respectivo sistema de arquivos, podendo esse ser local ou remoto (SILBERSCHATZ; GALVIN; GAGNE, 2008).

Nos *sistemas de arquivos locais*, como o EXT4 (MATHUR et al., 2007), ocorre o mapeamento entre os arquivos e os blocos nos dispositivos de armazenamento. Esse nível é responsável também pelo gerenciamento de *buffers* e *caches*: espaços designados na memória principal, que armazenam dados e metadados com o intuito de melhorar o desempenho das operações (SILBERSCHATZ; GALVIN; GAGNE, 2008).

A comunicação entre sistemas de arquivos e dispositivos de armazenamento, como *HDDs* e *SSDs*, é realizada por meio do *controle de E/S*. Esse controle consiste de *drivers* de dispositivos e mecanismos de interrupção que transferem as informações da memória principal para o dispositivo e vice-versa (SILBERSCHATZ; GALVIN; GAGNE, 2008). A principal função do controle de E/S é traduzir comandos de alto nível para as instruções específicas de cada dispositivo de armazenamento.

Sistemas de arquivos remotos, também conhecidos como *sistemas de arquivos distribuídos*, como o *Network File System* (NFS) (SANDBERG et al., 1985) e o OrangeFS (MOORE et al., 2011), apresentam para seus usuários funcionalidades similares aos sistemas de arquivos locais. A principal diferença está na sua arquitetura, em que operações e dados trafegam por uma *rede de interconexão* (COULOURIS et al., 2012). Detalhes sobre sistemas de arquivos distribuídos são apresentados na próxima seção.

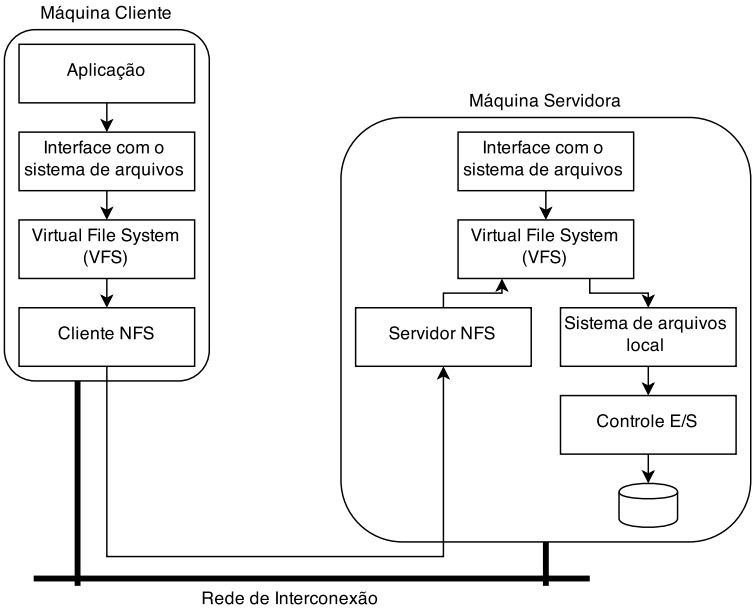
2.2 SISTEMAS DE ARQUIVOS DISTRIBUÍDOS

Sistemas de arquivos distribuídos permitem que processos armazenem e recuperem arquivos remotamente, por meio de uma rede de interconexão, exatamente como fariam em dispositivos locais (COULOURIS et al., 2012). A principal motivação para o uso de sistemas de arquivos distribuídos é o compartilhamento de arquivos entre processos, que podem, inclusive, estar executando em diferentes recursos computacionais (COULOURIS et al., 2012). O compartilhamento de arquivos nesses sistemas é facilitado pelo provimento de um espaço de nomes

global. Segundo Tanenbaum e Steen (2006), os sistemas de arquivos distribuídos podem ser classificados, de acordo com a sua organização, como: cliente-servidor, baseado em *cluster* e simétricos.

Arquiteturas *cliente-servidor*, como o NFS (SANDBERG et al., 1985), são caracterizadas pela centralização do armazenamento no lado do servidor (TANENBAUM; STEEN, 2006). Nessas arquiteturas, um processo cliente se comunica por meio da rede de interconexão com um processo servidor localizado em outra máquina, conforme ilustrado na Figura 2. Apesar de oferecer uma solução simples e amplamente utili-

Figura 2: Arquitetura básica do NFS para sistemas UNIX e Linux.

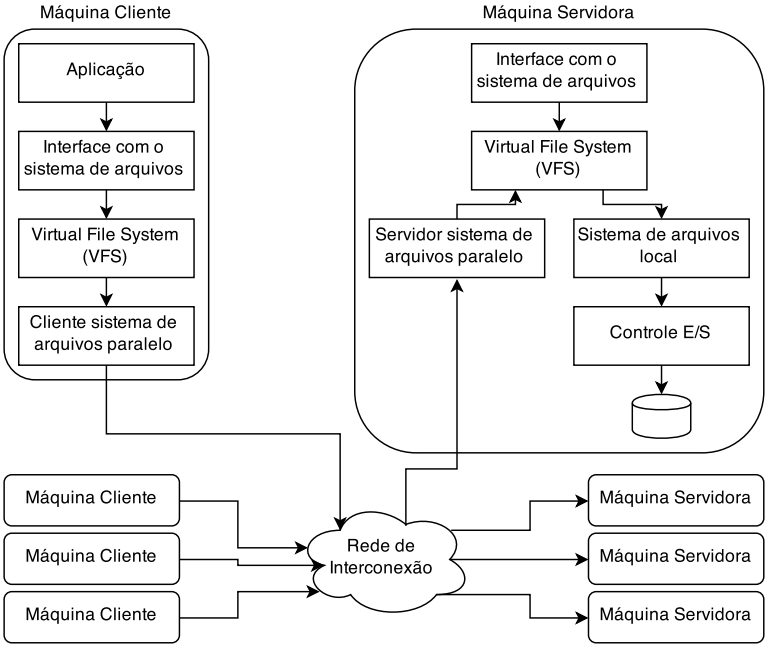


Fonte: Baseado em Tanenbaum e Steen (2006).

zada para compartilhamento de arquivos, arquiteturas cliente-servidor tendem a apresentar baixo desempenho quando muitos acessos são realizados concorrentemente (LATHAM; ROSS, 2013).

Sistemas de arquivos *baseados em cluster*, também conhecidos como *sistemas de arquivos paralelos*, apresentam arquitetura similar à cliente-servidor, como pode ser observado na Figura 3. A principal diferença se dá pela utilização de um aglomerado (*cluster*) de recursos computacionais atuando como servidores de armazenamento. Uma

Figura 3: Arquitetura básica de sistemas de arquivos paralelos.



característica particular dos sistemas de arquivos paralelos é o particionamento e a distribuição de arquivos entre os múltiplos servidores (TANENBAUM; STEEN, 2006). Outra característica é a segregação das funções de gerenciamento de dados e de metadados. Exemplos desses sistemas são o Lustre (BRAAM; SCHWAN, 2002), GPFS (SCHMUCK; HASKIN, 2002) e o PVFS (LIGON III; ROSS, 1996) (atualmente desenvolvido sob o nome OrangeFS). Sistemas de arquivos paralelos são vistos com mais detalhes na próxima seção.

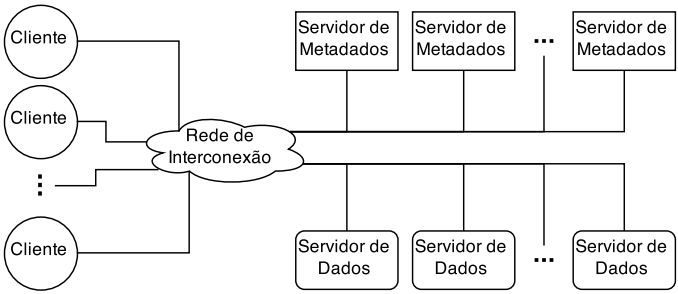
Organizações *simétricas*, como o sistema de arquivos *peer-to-peer* Ivy (MUTHITACHAROEN et al., 2002), correspondem a arquiteturas descentralizadas (TANENBAUM; STEEN, 2006), ou seja, não existe a figura de um elemento central responsável pelo gerenciamento do sistema de arquivos. A implementação desses sistemas consiste basicamente da distribuição dos arquivos entre múltiplos servidores utilizando tabelas *hash* distribuídas, combinada à mecanismos de busca baseados em chave (TANENBAUM; STEEN, 2006).

2.2.1 Sistemas de Arquivos Paralelos

Sistemas de arquivos paralelos vêm sendo amplamente utilizados em ambientes de computação de alto desempenho (SHAN; ANTYPAS; SHALF, 2008; SAINI et al., 2012; SOARES et al., 2012b). Por meio da agregação de recursos computacionais, como dispositivos de armazenamento, links de rede e memória principal, os sistemas de arquivos paralelos conseguem prover configurações com ampla capacidade de armazenamento e alta vazão (em termos de operações por unidade de tempo).

Apesar de variações serem encontradas nas diferentes implementações, os sistemas de arquivos paralelos consistem basicamente de três componentes: servidor de dados, servidor de metadados e cliente (SHAN; ANTYPAS; SHALF, 2008; BARRO et al., 2002). A Figura 4 ilustra esses componentes.

Figura 4: Visão sistemática dos sistemas de arquivos paralelos.



O *servidor de dados* é o componente responsável pela persistência do conteúdo dos arquivos. Sua função é basicamente interagir com o(s) dispositivo(s) de armazenamento controlado(s) por ele para armazenar e recuperar arquivos. A quantidade máxima de servidores de dados possíveis para uma configuração depende, geralmente, da implementação do sistema de arquivos paralelo. Essa limitação, entre outros aspectos, determina a capacidade máxima de armazenamento do sistema e o nível de paralelização atingível.

Manter atualizadas as informações referentes aos arquivos (metadados) é função do *servidor de metadados*. Isso inclui não apenas a manutenção de dados como nome e permissões, mas também a estrutura de diretórios como um todo. Algumas implementações de sistemas de arquivos paralelos, como o Ceph (WEIL et al., 2006) e o Lustre

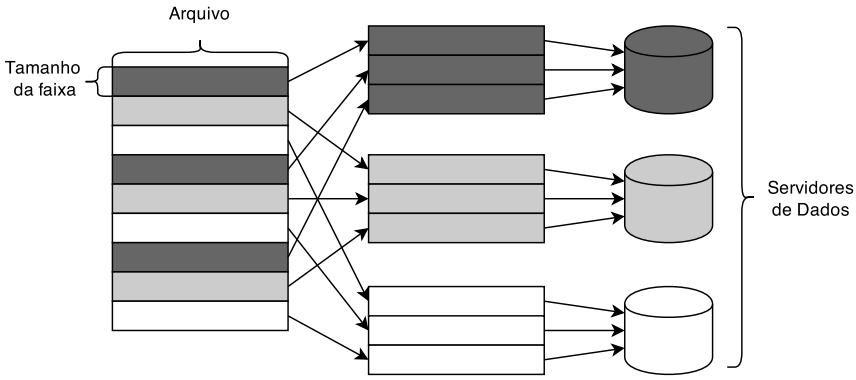
(BRAAM; SCHWAN, 2002), suportam apenas um servidor de metadados na sua configuração. Outros, como o OrangeFS (MOORE et al., 2011) e o GPFS (SCHMUCK; HASKIN, 2002), permitem a distribuição dos metadados entre múltiplos servidores.

O *cliente* é o componente que possibilita a interação com o sistema de arquivos paralelo. Esse componente está usualmente localizado no nodo de computação e é utilizado pelas aplicações para a realização das operações sobre o sistema de arquivos. Em geral, esses clientes oferecem, além de uma API (sigla do inglês, *Application Programming Interface*) nativa, suporte para o padrão POSIX (CARNS et al., 2000; BRAAM, 2004; SCHMUCK; HASKIN, 2002).

Vale observar como uma característica geral dos sistemas de arquivos paralelos a separação entre clientes e servidores. Por outro lado, é comum observar os papéis de servidor de dados e de metadados sendo realizados pelo mesmo nodo computacional (CARNS et al., 2009).

Como mencionado anteriormente, uma das principais características dos sistemas de arquivos paralelos é o particionamento e a distribuição do conteúdo dos arquivos entre múltiplos servidores. Esse processo é chamado de divisão de arquivo (do inglês, *file striping*) (TANENBAUM; STEEN, 2006). A Figura 5 apresenta um exemplo do processo para um arquivo com 9 unidades de tamanho e um sistema de arquivos paralelo com 3 servidores de dados.

Figura 5: Processo de divisão de arquivos (*file striping*).



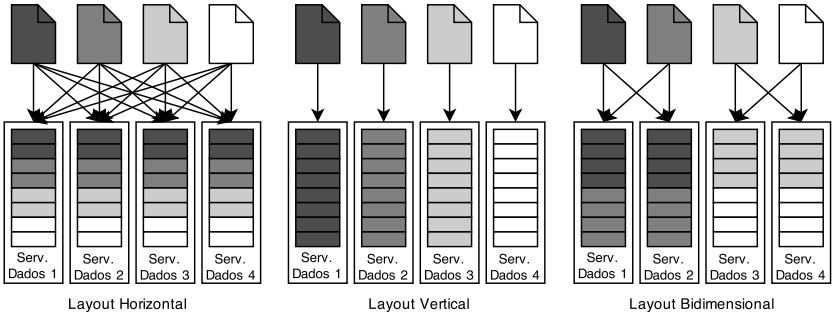
O processo consiste em dividir o arquivo em segmentos consecutivos, ou faixas (do inglês, *stripes*), e distribuir esses segmentos entre os servidores de dados. O tamanho das faixas são em geral configuráveis

no sistema de arquivos paralelo, podendo ser otimizadas para a carga de trabalho em questão.

A ideia por trás desse processo é bastante simples: distribuindo um arquivo entre múltiplos servidores torna possível realizar as operações de armazenamento e de recuperação de forma paralela (TANENBAUM; STEEN, 2006). Estudos anteriores demonstraram que o processo é mais eficiente para grandes arquivos, em que os benefícios da paralelização superam os custos de comunicação (HILDEBRAND; WARD; HONEYMAN, 2006; CARNS et al., 2009; LI et al., 2011).

A distribuição das faixas de um arquivo pode acontecer de diferentes formas, chamadas layouts de distribuição. Os layouts comumente encontrados são: horizontal, vertical e bidimensional (SONG et al., 2012). Na Figura 6, exemplos desses layouts são ilustrados.

Figura 6: Layouts de distribuição em sistemas de arquivos paralelos.



Fonte: Baseado em Song et al. (2012).

Na distribuição *horizontal*, também conhecida como *round robin*, as faixas de um arquivo são distribuídas entre todos os servidores de dados. Esse layout possibilita extrair o máximo de paralelização do ambiente.

A distribuição *vertical* consiste em manter todos as faixas de um arquivo em um mesmo nodo de armazenamento, de forma que cada nodo armazena o conteúdo completo de um conjunto de arquivos. Nesse layout, operações sobre um mesmo arquivo não são paralelizáveis.

O layout de distribuição *bidirecional* é um híbrido dos anteriores. Nesse layout, as faixas de um arquivo são distribuídos entre um subconjunto dos nodos de armazenamento.

2.3 CARACTERIZAÇÃO E MODELAGEM

Caracterização de carga de trabalho é um processo no qual modelos são construídos para descrever e reproduzir o comportamento da carga de trabalho de um sistema computacional (ELNAFFAR; MARTIN, 2006). Compreender as características da carga de trabalho dos sistemas é essencial para o projeto de arquiteturas computacionais eficientes (OBAIDAT; BOUDRIGA, 2010).

A análise da carga de trabalho tem um papel fundamental em todos os estudos em que índices de desempenho são determinados (CALZAROSSA; SERAZZI, 1993). Segundo Obaidat e Boudriga (2010), três técnicas podem ser utilizadas para caracterizar o desempenho de um sistema computacional: modelagem analítica, simulação e mensuração e teste.

Técnicas de *modelagem analítica* são usualmente as mais simples, uma vez que as expressões matemáticas são obtidas de maneira rápida e a execução apresenta baixo custo computacional. Em contrapartida, as estimativas obtidas com modelos analíticos são sempre aproximações (OBAIDAT; BOUDRIGA, 2010). Modelos analíticos são particularmente úteis para problemas em que uma resposta em um curto período de tempo é desejada, mesmo que essa resposta apresente um erro relativo aceitável.

Simulações oferecem uma maior acurácia nas estimativas, além de permitir uma análise passo-a-passo do comportamento do sistema (OBAIDAT; BOUDRIGA, 2010). Contudo, sua construção é mais complexa e o tempo necessário para sua execução tende a ser mais elevado que nos modelos analíticos. São especialmente interessantes para investigar situações pontuais no comportamento de um sistema, por meio de uma análise das etapas executadas. Na área de sistemas de arquivos paralelos, exemplos de simuladores são o IMPIOUS (MOLINA-ESTOLANO et al., 2009), HECIOS (SETTLEMYER, 2009), PFSsim (LIU; FIGUEIREDO, 2011) e CODES (LIU et al., 2012).

Entre as três técnicas descritas, a mensuração e teste, sobre um protótipo ou ambiente real, é a que produz a maior acurácia. No entanto, é também a mais custosa, tanto em termos temporais como financeiros, pois depende da existência de um ambiente experimental. Essa característica torna sua utilização inviável nas etapas iniciais de projetos de ambientes e sistemas computacionais (OBAIDAT; BOUDRIGA, 2010).

No contexto de sistemas de arquivos paralelos, uma das principais dificuldades encontradas na caracterização e modelagem do desem-

penho é a quantidade de variáveis envolvidas no processo. Enquanto em sistemas de arquivos locais, o desempenho é predominantemente dependente da quantidade de dados sendo manipulada (NGUYEN; APON, 2011), em sistemas de arquivos paralelos outros fatores como a rede de interconexão e a quantidade de servidores de dados também afetam significativamente o desempenho do sistema. A análise desse tipo de problema, em que o comportamento de uma variável, chamada *variável dependente*, depende do comportamento de outras variáveis, chamadas *variáveis independentes*, é chamada de análise multivariada (EVERITT; HOTHORN, 2011).

2.4 CONSIDERAÇÕES

A medida que as aplicações crescem em escala, em que aumentam não apenas o número de processos fazendo acessos simultâneos, mas também o volume de dados manipulados, fica evidente que sistemas de arquivos locais ou até mesmo soluções distribuídas com arquiteturas centralizadas (cliente-servidor) não são capazes de suprir o nível de concorrência e de vazão desejados. Para atender essa demanda, arquiteturas distribuídas baseadas em *cluster*, ou sistemas de arquivos paralelos, vêm sendo amplamente utilizadas.

Sistemas de arquivos paralelos oferecem alta capacidade de armazenamento com alta vazão por meio da agregação de recursos computacionais, como dispositivos de armazenamento, links de rede e memórias principais. Por meio da técnica de divisão de arquivos (*file striping*), em que um arquivo é particionado e suas partes são distribuídas entre múltiplos servidores de dados, as operações de armazenamento e recuperação podem ser realizadas de forma paralela.

Como observado em um estudo anterior (INACIO; DANTAS, 2014), a caracterização e modelagem do desempenho de sistemas computacionais possibilita a proposta de soluções de otimização mais eficientes. Contudo, a quantidade de variáveis influenciando o desempenho dos sistemas de arquivos paralelos torna a caracterização e modelagem desses sistemas uma tarefa desafiadora. Técnicas de análise multivariada são necessárias para abordar problemas como esse, onde o comportamento de uma variável depende não apenas de outra, mas sim de várias outras variáveis combinadas.

No próximo capítulo são discutidas propostas de soluções encontradas na literatura para o problema de modelagem do desempenho de sistemas de arquivos paralelos.

3 TRABALHOS RELACIONADOS

O problema da caracterização e modelagem do desempenho de sistemas de arquivos paralelos foi abordado em alguns trabalhos anteriores. O processo utilizado para coleta de trabalhos relacionados com a presente pesquisa é apresentado na Seção 3.1. Na Seção 3.2, os trabalhos encontrados na literatura são apresentados e discutidos. A Seção 3.3 sumariza a discussão desse capítulo pontuando oportunidades de melhoria nas propostas de modelos de desempenho existentes até o momento.

3.1 REVISÃO SISTEMÁTICA

Uma revisão sistemática da literatura (RSL), seguindo os princípios propostos por Kitchenham (2004), foi realizada visando identificar propostas de modelos e caracterizações de desempenho para sistemas de arquivos paralelos. Foram utilizadas no levantamento as bases de dados *IEEEXplore* (IEEE, 2014), *ScienceDirect* (ELSEVIER, 2014a), *Scopus* (ELSEVIER, 2014b) e *Web of Science* (THOMSON REUTERS, 2014). Os critérios de seleção de documentos consideraram combinações das seguintes palavras-chave e algumas variações (ex.: singular e plural): *performance*, *workload*, *characterization*, *model*, *parallel*, *distributed*, *network*, *file system* e *storage*. Também foram incluídos nos critérios nomes e siglas de implementações de sistemas de arquivos paralelos, como *PVFS*, *GPFS* e *Lustre*.

Com base nesses critérios, foram localizados no total 85 trabalhos de pesquisa. Sobre esses trabalhos foi aplicado um processo manual de seleção. Primeiramente foram excluídos documentos duplicados, apresentações de conferências, editoriais e resumos. Em seguida, documentos não relacionados ao assunto da pesquisa foram desconsiderados. Por fim, após a leitura e classificação dos trabalhos, seis foram identificados como altamente relevantes com a presente pesquisa.

Vale destacar que durante a realização deste trabalho, outros documentos não originados da RSL também foram considerados. Embora um número significativo de trabalhos tenham sido encontrados, apenas os seis trabalhos resultantes do processo de RSL apresentaram as características desejadas nesta pesquisa, como: formalização do modelo, avaliação de acurácia, entre outros. Desta forma, considera-se que os artigos discutidos na seção a seguir representam o estado da arte em

modelagem de desempenho de sistemas de arquivos paralelos.

3.2 DISCUSSÃO

O trabalho de Sun, Chen e Yin (2009), estendido por Song et al. (2011), propõe a utilização de um modelo analítico para orientar a seleção de layouts de distribuição de dados no PVFS2 e, com isso, obter um melhor desempenho nas aplicações. O modelo utilizado considera nove parâmetros: número de nodos de computação, número de nodos de armazenamento, número de processos em um nodo de computação, tamanho do dado, tempo para estabelecimento de uma conexão na rede, tempo para transmitir uma unidade do dado na rede, latência de uma operação local de E/S no nodo de armazenamento, tempo para ler/escrever uma unidade do dado em disco e o número de grupos de armazenamento (para uma distribuição bidimensional). A avaliação da proposta foi feita utilizando um conjunto de experimentos em ambientes reais, desenvolvido com os benchmarks IOR (LAWRENCE LIVERMORE NATIONAL LABORATORY, 2014) e mpi-tile-io (ARGONNE NATIONAL LABORATORY, 2002). A acurácia do modelo utilizado não é apresentada quantitativamente. Pelos gráficos apresentados, é possível apenas observar que o modelo subestima o comportamento real, com considerável erro absoluto, em mais da metade dos cenários. Parte desse erro absoluto é atribuído pelos autores ao efeito da cache nos resultados experimentais, efeito este não considerado no modelo. Pode ser considerado também que a simplicidade do modelo, por exemplo, na modelagem dos aspectos da rede de interconexão, contribua também significativamente para a inacurácia das predições. Vale ressaltar que, mesmo nessas condições, melhorias de desempenho entre 13% e 74% foram obtidas pela seleção do layout de distribuição orientado ao modelo.

Uma abordagem similar foi utilizada por Lu et al. (2012). Nesse trabalho, um modelo analítico é utilizado para direcionar otimizações em peneiramento de dados (do inglês, *data sieving*) no PVFS2. O peneiramento de dados é uma técnica que agrupa requisições de E/S pequenas e não contíguas com o objetivo de reduzir a latência desse tipo de requisições (THAKUR et al., 1996). O modelo estima o tempo de leitura/escrita no sistema de arquivos paralelo como uma função do tempo de transmissão na rede, do tempo para leitura/escrita de dados em disco e da latência para atendimento de requisições de E/S pendentes. São considerados o número de processos executando E/S

em um nodo de computação, o número de nodos de armazenamento, o tempo para estabelecimento de um conexão de rede, o tempo para transmissão de uma unidade de dado, o tempo para leitura e escrita de uma unidade de dado em disco, tempo de espera das requisições na fila de E/S e o tamanho das requisições de leitura e escrita. O trabalho não apresenta uma avaliação da acurácia do modelo. São apresentados, apenas graficamente, resultados indicando que o uso da abordagem proposta proporciona um redução no tempo de resposta das aplicações.

Nguyen e Apon (2012) propõem um modelo de desempenho, baseado em simulação, para o PVFS2 utilizando redes de Petri coloridas. O modelo é composto por submodelos de leitura, escrita, servidor, cliente e rede de comunicação. Experimentos com quatro nodos de armazenamento e tamanhos de blocos de dados entre 8 KB e 2 MB, apresentaram erros relativos de aproximadamente 20%, para um cliente, e maiores ou iguais a 40% para quatro ou mais clientes. Segundo os autores, o motivo para o aumento do erro relativo com múltiplos clientes está relacionado a não consideração de fatores complexos da rede de comunicação no modelo. Redes de Petri coloridas são particularmente interessantes quando o objetivo é avaliar passo-a-passo o comportamento de um sistema (OBAIDAT; BOUDRIGA, 2010). As características das redes de Petri facilitam a representação de aspectos de bloqueio e sincronização. Em contrapartida, a complexidade desse tipo de representação, em comparação com modelos analíticos, dificulta a observação dos relacionamentos entre os parâmetros que colaboram para a determinação do desempenho do sistema de maneira geral.

No trabalho de Zhao et al. (2010) é apresentado um modelo analítico de desempenho para o sistema de arquivos paralelo Lustre. O modelo foi construído utilizando árvores de classificação e regressão, e permite estimar a vazão, em termos de fluxo de dados (MB/s) e de operações de E/S (IO/s) por unidade de tempo, e a latência do sistema. Um amplo conjunto com mais de 20 parâmetros é considerado pelo modelo. A abordagem utilizada nessa proposta é a estimativa do comportamento de um ambiente baseado no comportamento de outro. Essa correlação entre o desempenho dos ambientes é obtida por meio do treinamento de uma função Φ, que considera as características da carga de trabalho e do consumo dos dois ambientes. Experimentos conduzidos para oito casos de teste apresentaram erros relativos entre 17% e 28%. Uma desvantagem dessa abordagem é sua dependência de informações de pelo menos um ambiente real, para que outro similar possa ser avaliado. Outra questão que vale ressaltar é que parâmetros especí-

ficos do Lustre foram considerados na construção desse modelo, o que pode dificultar sua adaptação ou reduzir sua acurácia para utilização na avaliação de outros sistemas de arquivos paralelos sem características correspondentes.

Zhao e Hu (2010) propõem a utilização de análise relacional cinza para estimar a vazão do sistema. Segundo os autores, um diferencial da abordagem adotada é uma menor necessidade de dados brutos para a estimativa do comportamento do sistema. Uma avaliação experimental do modelo analítico proposto foi conduzida para o sistema de arquivos paralelo Lustre, considerando cinco parâmetros: número de *Object Storage Servers* (OSS), tipo de *journaling* do sistema de arquivos, tipo de disco, abordagem de conexão do armazenamento e número de *threads* por *Object Storage Target* (OST). Os resultados mostraram erros relativos entre 4% e 11% para oito casos de teste. A abordagem proposta nessa pesquisa não define exatamente uma relação entre fatores de sistemas de arquivos paralelos e seus efeitos no desempenho do sistema. A proposta consiste da utilização de uma técnica para a verificação de correlacionamento entre parâmetros de forma geral, o que possibilita estimar um parâmetro com relação a outros. Essa característica possibilita a utilização dessa abordagem para problemas de identificação de limitações de desempenho em sistemas de arquivos paralelos. Contudo, a abordagem proposta não é adequada para, por exemplo, avaliações de diferentes cenários visando projetos de arquiteturas de armazenamento.

3.3 CONSIDERAÇÕES

A revisão sistemática da literatura conduzida nesse trabalho identificou a existência de algumas propostas de modelos para representação do desempenho para sistemas de arquivos paralelos. As características destes modelos, apresentadas e discutidas nesse capítulo, estão sumarizadas na Tabela 1.

A partir do estudo destes trabalhos, observam-se alguns aspectos e oportunidades de melhoria na representação do desempenho do armazenamento em sistemas de arquivos paralelos que serão exploradas nessa pesquisa.

Um dos aspectos observados neste estudo foi o compromisso entre granularidade e acurácia. Verificou-se que representações mais detalhadas, como o modelo baseado em simulação de Nguyen e Apon (2012), ou baseadas em técnicas de regressão, como as propostas de Zhao et al. (2010) e Zhao e Hu (2010), apresentaram menores erros relativos. Con-

Tabela 1: Sumário dos trabalhos relacionados à presente presquisa.

Trabalho	**Técnica**	**SAP**	**Erro**	**Prós**	**Contras**
Sun, Chen e Yin (2009), Song et al. (2011)	Modelagem analítica	PVFS2	N.I.	Baixo custo de execução; variáveis comuns	Características não exploradas; suposta baixa acurácia
Lu et al. (2012)	Modelagem analítica	PVFS2	N.I.	Baixo custo de execução; variáveis comuns	Características não exploradas; suposta baixa acurácia
Nguyen e Apon (2012)	Baseado em simulação	PVFS2	20-40%	Alta granularidade; análise temporal	Baixa acurácia com múltiplos clientes
Zhao et al. (2010)	Modelagem analítica	Lustre	17-28%	Alta acurácia	Dependência de dados reais; especificidade de parâmetros
Zhao e Hu (2010)	Modelagem analítica	Lustre	4-11%	Alta acurácia	Dependência de dados reais; relação entre variáveis

tudo, as formalizações propostas nestes trabalhos não deixam explícito a influência das variáveis consideradas no desempenho do sistema de arquivos paralelo. Além disso, a dependência de informações de desempenho de ambientes reais para a realização de estimativas restringe a utilização do modelo.

A abordagem de representação analítica utilizada nos trabalhos de Sun, Chen e Yin (2009), Song et al. (2011) e Lu et al. (2012), por outro lado, deixa explícito o relacionamento entre as variáveis consideradas no modelo e como elas interferem no desempenho do sistema. Adicionalmente, a generalidade do conjunto de variáveis considerados na representação permite que o modelo seja utilizado para diferentes cenários de uso. Entretanto, características fundamentais de sistemas de arquivos paralelos, como o tamanho da faixa, ou com impacto significativo no desempenho, como a cache, não foram considerados.

Além do efeito da cache, outro efeito apontado pelos autores como uma das principais fontes de desvios nas estimativas dos modelos foi o efeito da rede de interconexão. A simplificação adotada no trabalho de Nguyen e Apon (2012), por exemplo, apresenta erros relativos superiores a 40% para cenários com múltiplos clientes. Isso se deve a dificuldade na representação de aspectos como contenção e congestionamento, comuns em cenários com nível significativo de concorrência de acesso.

É importante destacar também a abrangência destas propostas. Embora a avaliação destes modelos, quando apresentada, tenha con-

siderado a comparação com resultados obtidos em ambientes reais, de maneira geral, um pequeno conjunto de cenários foi considerado em cada trabalho. Desta forma, suspeita-se que uma avaliação mais abrangente possa indicar um acurácia diferente das apresentadas.

O próximo capítulo apresenta a proposta dessa dissertação, detalhando um modelo analítico multivariado, construído e avaliado a partir de um extenso conjunto de experimentos conduzidos em diferentes plataformas computacionais reais, capaz de representar o impacto de um grande número de variáveis no desempenho do armazenamento em sistemas de arquivos paralelos, incluindo variáveis e efeitos não explorados em trabalhos anteriores, para um espectro mais abrangente de cenários de uso.

4 PROPOSTA

Como observado nos capítulos anteriores, o desempenho de sistemas de arquivos paralelos depende de vários fatores. Compreender o relacionamento entre esses fatores é importante para explorar ao máximo o desempenho desses sistemas, principalmente quando se considera a enorme demanda de armazenamento prevista para o futuro próximo. Esse trabalho propõe um modelo analítico multivariado para representar o desempenho do armazenamento em sistemas de arquivos paralelos diante de diferentes configurações e cargas de trabalho.

A opção por uma modelagem analítica visa prover uma aproximação em um curto período de tempo e com baixo custo computacional. Essas características permitem que o modelo seja aplicado à diferentes situações: desde soluções de otimização em algoritmos (ex.: escalonamento) até projetos de novas arquiteturas de armazenamento. Prevendo essas possibilidades, o modelo proposto foi construído de forma a não necessitar de informações reais de desempenho, bastando o fornecimento de alguns parâmetros de entrada.

O aspecto multivariado advém de o desempenho do sistema de arquivos paralelo depender do relacionamento entre múltiplos fatores. A representação desse relacionamento é desafiadora não apenas pela quantidade de variáveis envolvidas, mas também pela característica de algumas dessas variáveis. Por exemplo, a representação da rede de interconexão, como visto no capítulo anterior, é um dos principais obstáculos na modelagem de sistemas de arquivos paralelos, apesar de algumas simplificações para esse aspecto específico existirem na literatura (BERTSEKAS; GALLAGER, 1992; PINKSTON; DUATO, 2006; VELHO et al., 2013).

Uma das contribuições dessa pesquisa está na representação diferenciada do efeito de um amplo conjunto de variáveis no desempenho de sistemas de arquivos paralelos. Optou-se nesse trabalho, por considerar apenas variáveis gerais da arquitetura básica dos sistemas de arquivos paralelos. Essa abordagem faz com que o modelo proposto não fique atrelado a uma determinada implementação ou arquitetura, podendo ser utilizado para avaliar dferentes sistemas e cenários de uso. Embora características específicas tenham sido evitadas, a presente pesquisa engloba um conjunto significativo de variáveis, explorando, inclusive, características não abordadas ou cujo efeito no comportamento dos sistemas de arquivos paralelos não foi explicitado em trabalhos anteriores.

Os fatores, ou variáveis, abordados nesse trabalho são discutidos

na Seção 4.1. A Seção 4.2 apresenta a formalização do modelo. Considerações finais sobre a proposta desse trabalho são apresentadas na Seção 4.3.

4.1 FATORES

Como discutido na Seção 2.2.1, sistemas de arquivos paralelos são basicamente compostos por três componentes: servidor de dados, servidor de metadados e clientes. Na prática, o que se observa nos sistemas de arquivos paralelos atuais é o papel de servidor de dados e de metadados sendo desempenhado pelo mesmo recurso computacional, ou seja, os serviços responsáveis pelo gerenciamento dos dados e dos metadados são hospedados em um mesmo nodo computacional, o qual referenciamos nessa dissertação como *nodo de armazenamento*.

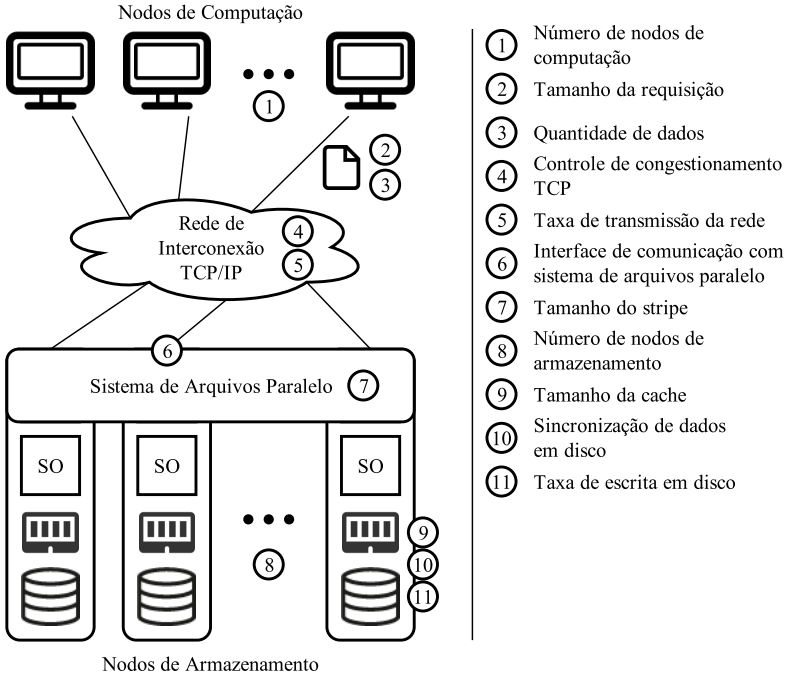
A Figura 7 apresenta a arquitetura básica de sistemas de arquivos paralelos considerada para a elaboração do modelo proposto nessa dissertação. Na figura são destacados os fatores avaliados e suas relações com os componentes da arquitetura básica.

A escolha desse conjunto de fatores se deu de forma incremental. Um primeiro conjunto de fatores foi derivado de conceitos fundamentais obtidos por meio da revisão da literatura. Vale destacar a inclusão do tamanho da faixa (*stripe*) nesse conjunto, característica esta fundamental em sistemas de arquivos paralelos e não explorada nas propostas de modelos analíticos encontrados na literatura até então. Outros fatores, como o tamanho da cache dos nodos de armazenamento e a sincronização de dados, foram incluídos durante a análise experimental. Verificou-se que o impacto destas variáveis no desempenho dos sistemas de arquivos paralelos é significativo, o que justifica sua consideração no modelo proposto nesta dissertação.

Para fins de representação do desempenho do armazenamento em sistemas de arquivos paralelos, conforme objetivo desta dissertação, considera-se que o conjunto de fatores selecionados é suficiente. Embora outros fatores, como custos de computação, pudessem ser considerados, entende-se que a inclusão destes poderia vincular o modelo a algum tipo de implementação, o que vem de encontro a um dos objetivos desta pesquisa, que é permitir a utilização do modelo para diferentes configurações e cargas de trabalho.

Para facilitar a compreensão da influência desses fatores no desempenho do sistema de arquivos paralelo, nessa dissertação, estes foram divididos em dois grupos: características de carga de trabalho e

Figura 7: Representação da arquitetura básica de sistemas de arquivos paralelos, com destaque para os fatores abordados nessa dissertação.



fatores ambientais.

4.1.1 Características de Carga de Trabalho

Neste trabalho, foram considerados como *características de carga de trabalho* os fatores que interferem no volume de trabalho imposto ao sistema de arquivos paralelo do ponto de vista das aplicações. Em outras palavras, referem-se a características determinadas ou inerentes às aplicações. Esse grupo compreende os seguintes fatores: número de nodos de computação, tamanho da requisição, quantidade de dados, interface de comunicação com o sistema de arquivos paralelo e a sincronização dos dados em disco.

O *número de nodos de computação* define a quantidade de cli-

entes simultâneos operando sobre o sistema de arquivos paralelo. Mais clientes escrevendo simultaneamente significa uma maior carga de trabalho tanto na rede de interconexão quanto nos nodos de armazenamento, dado que um volume maior de dados trafega pelo sistema.

O *tamanho da requisição* representa a quantidade de bits do dado que um cliente envia para o sistema de arquivos paralelo a cada requisição de uma operação de escrita. Quanto menor o tamanho da requisição, maior a quantidade de mensagens que serão necessárias para transmitir todo o dado para o sistema de arquivos. Isso, por sua vez, significa um maior atraso em função da comunicação de rede entre o nodo de computação e os nodos de armazenamento, o que resulta em um aumento no tempo de resposta do sistema.

Assim como ocorre em sistemas de arquivos locais, a *quantidade de dados* é um dos fatores com maior impacto no tempo de resposta da operação de escrita. Maiores volumes de dados vão consumir um tempo maior para serem escritos. Vale ressaltar que estudos anteriores (HILDEBRAND; WARD; HONEYMAN, 2006; CARNS et al., 2009; LI et al., 2011) mostraram que o desempenho de operações de escrita para pequenos volumes de dados apresentam menor desempenho em sistemas de arquivos paralelos, em função dos custos de comunicação de rede.

A *interface de comunicação com o sistema de arquivos paralelo* determina a semântica utilizada para a realização das operações de escrita. Interfaces de uso geral, como a interface POSIX, tendem a ser mais conservadoras no que diz respeito ao bloqueio e sincronização de arquivos. Interfaces desenvolvidas com foco em ambientes de alto desempenho, como a interface MPI-IO, oferecem semântica mais relaxada nesses aspectos. Quanto mais conservador o controle das operações da interface utilizada pelo cliente, maior o custo de cada operação.

A *sincronização de dados em disco* refere-se ao processo de persistência das operações de escrita realizadas em memória (cache de escrita) no disco rígido local. Essa operação é iniciada quando uma chamada de sistema `fsync()` é realizada pelo cliente. Como o tempo de acesso a memória é inferior ao tempo de acesso ao disco rígido, a realização da sincronização dos dados aumenta o tempo de resposta do sistema.

4.1.2 Fatores Ambientais

Foram considerados como *fatores ambientais* aspectos da arquitetura computacional e da configuração do sistema de arquivos paralelo.

Os seguintes fatores foram considerados nesse grupo: número de nodos de armazenamento, tamanho da faixa (*stripe*) do sistema de arquivos paralelo, tamanho da cache de escrita, algorítmo de controle de congestionamento TCP, taxa de transmissão da rede de interconexão e a taxa de transferência de dados para o disco local.

O *número de nodos de armazenamento* é um dos fatores que define o grau de paralelização da operação de escrita em sistemas de arquivos paralelos. No contexto geral, mais nodos de armazenamento representam um maior grau de paralelização, que por sua vez, implica em uma maior capacidade de vazão de dados. Além disso, um maior número de nodos de armazenamento oferece também uma maior capacidade de armazenamento para o sistema, tanto em disco rígido quanto em memória cache.

Outro fator que influencia no grau de paralelização de um sistema de arquivos paralelos é o *tamanho da faixa*. Quanto maior o tamanho da faixa, menor será o número de partes em que um dado será dividido. Isso significa um menor grau de paralelização e, por consequência, uma menor capacidade de vazão. Em uma operação de escrita, o número máximo de nodos de armazenamento que serão utilizados para persistência do dado depende da quantidade de partes em que esse dado for dividido.

O *tamanho da cache de escrita* é uma característica de cada nodo de armazenamento. Corresponde ao espaço em memória principal disponibilizado pelo nodo de armazenamento para cache das operações de escrita. O valor desse parâmetro depende do tamanho da memória principal e das configurações de cache. Nos sistemas Linux, essas configurações são realizadas por meio de parâmetros do *kernel*, como `dirty_background_ratio` e `dirty_ratio`. Mais espaço de cache de escrita significa suportar uma maior quantidade de operações de escrita em memória, que possui uma velocidade de acesso muito superior à dos discos rígidos atuais.

O *algoritmo de controle de congestionamento TCP* atua sobre o tamanho da janela de transmissão TCP para controlar a taxa de transmissão e o congestionamento na rede de interconexão. O congestionamento na rede provoca a perda e reenvio de dados dos nodos de computação para os nodos de armazenamento. Logo, em cenários com elevadas cargas de trabalho, sejam elas provocadas pela quantidade de clientes simultâneos ou pelo volume de dados sendo escrito concorrentemente, diferentes técnicas de controle de congestionamento tendem a influenciar de forma diferente no desempenho do sistema.

A *taxa de transmissão da rede de interconexão* determina o tempo

necessário para transmitir um dado do nodo de computação para o nodo de armazenamento em uma operação de escrita em sistemas de arquivos paralelos. Quanto maior a taxa de transmissão, menor o tempo necessário para transmitir o dado pela rede.

De maneira similar, a *taxa de transferência de dados para o disco local* determina o tempo necessário para persistir um dado recebido pelo nodo de armazenamento no disco rígido local. Maiores taxas de transferência representam menores tempos de escrita.

4.2 MODELO ANALÍTICO MULTIVARIADO

O modelo proposto nesse trabalho tem por objetivo representar o desempenho do armazenamento em sistemas de arquivos paralelos diante de diferentes configurações e cargas de trabalho. Na seção anterior, os fatores que foram considerados para caracterizar essas configurações e cargas de trabalho foram apresentados. Uma combinação de valores para os diferentes fatores foi definida nesse trabalho como um *cenário de uso*. É importante observar que o foco do modelo é representar o desempenho geral do sistema de arquivos paralelo e não o desempenho de cada cliente ou processo operando no sistema.

No contexto desse trabalho, um cenário de uso consiste de um conjunto de n nodos de computação, $\mathcal{C} = \{c_1, c_2, c_3, \ldots, c_n)\}$, acessando simultaneamente, por meio de uma rede de interconexão, um sistema de arquivos paralelo composto por m nodos de armazenamento, $\mathcal{A} = \{a_1, a_2, a_3, \ldots, a_m\}$. Cada nodo de computação c_i, com i no intervalo $[1, n]$, escreve $d_c(i)$ bits em um arquivo no sistema de arquivos paralelos utilizando requisições de tamanho $r(i)$. Logo, a quantidade de requisições, $N_{rc}(i)$, submetidas pelo nodo de computação c_i é dada pela Equação (4.1).

$$N_{rc}(i) = \left\lceil \frac{d_c(i)}{r(i)} \right\rceil \tag{4.1}$$

Para prover maior vazão de escrita, o sistema de arquivos paralelo particiona e distribui os dados recebidos dos n nodos de computação entre os m nodos de armazenamento (ver Seção 2.2.1). Primeiramente, os dados de cada nodo de computação c_i, $d_c(i)$, são divididos em $N_{sc}(i)$ faixas de tamanho s. Por definição, a quantidade de faixas provenientes do nodo de computação c_i é dada pela Equação (4.2).

$$N_{sc}(i) = \left\lceil \frac{d_c(i)}{s} \right\rceil \tag{4.2}$$

Com base na quantidade de faixas $N_{sc}(i)$, na quantidade de nodos de armazenamento disponíveis m e no layout de distribuição do sistema de arquivos paralelo, é possível determinar a quantidade de nodos $M(i)$ que serão utilizados para armazenar os $d_c(i)$ bits enviados pelo nodo de computação c_i. Para os três layouts de distribuição básicos (horizontal, vertical e bidimensional), o valor de $M(i)$ é dado pela Equação (4.3).

$$M(i) = \begin{cases} \min\{N_{sc}(i), m\} & \text{, horizontal} \\ 1 & \text{, vertical} \\ \min\{N_{sc}(i), g\} & \text{, bidimensional} \end{cases} \tag{4.3}$$

em que g é o limite máximo de nodos de armazenamento configurado no sistema de arquivos paralelo para uma distribuição bidimensional.

Cada nodo de armazenamento a_j, com j no intervalo $[1, m]$, recebe, em um cenário de uso, $d_a(j)$ bits de dados. Essa quantidade de dados depende diretamente da quantidade de faixas $N_{sc}(i)$ enviadas por cada nodo de computação c_i e inversamente da quantidade $M(i)$ de nodos de armazenamento que serão utilizados por cada c_i. Supondo que o nodo de armazenamento a_j esteja entre os $M(k)$ nodos de armazenamento que serão utilizados por um subconjunto $\mathcal{C}'$ de $\mathcal{C}$, com k nodos de computação ($k \leq n$), a quantidade de dados $d_a(j)$ recebidas é dada pela Equação (4.4).

$$d_a(j) = s \times \sum_{k \in \mathcal{C}'} \frac{N_{sc}(k)}{M(k)} \tag{4.4}$$

A formação do subconjunto $\mathcal{C}'$ depende de como o sistema de arquivos paralelo implementa a seleção dos nodos de armazenamento e foge ao escopo desse trabalho.

De fato, para cenários onde $M(i)$ é menor que m, surge um nível de incerteza sobre o comportamento de diversas variáveis do sistema. Essas incertezas advém do possível desbalanceamento provocado pela seleção dos $M(i)$ nodos de armazenamento para atender cada c_i nodo de computação. Dependendo da seleção realizada, condições de desempenho muito diferentes podem ser observadas. Alternativas baseadas em análise combinatória e probabilidades foram abordadas nessa proposta para aproximar esses casos. Porém, essas alternativas não se mostraram escaláveis, visto que com o aumento da quantidade de nodos de computação e de armazenamento o número de possibilidades cresce exponencialmente.

Desse ponto em diante da proposta, serão considerados cenários balanceados. Em outras palavras, supõe-se que todos os nodos de armazenamento são utilizados por todos os nodos de computação ($M(i) = m$) e que os n nodos de computação enviam a mesma quantidade de dados, d_c, utilizando o mesmo tamanho de requisição r. Portanto, as Equações (4.1), (4.2) e (4.4) podem ser simplificadas, resultando, respectivamente, nas Equações (4.5), (4.6) e (4.7).

$$N_{rc} = \left\lceil \frac{d_c}{r} \right\rceil \tag{4.5}$$

$$N_{sc} = \left\lceil \frac{d_c}{s} \right\rceil \tag{4.6}$$

$$d_a = s \times n \times \left\lceil \frac{N_{sc}}{m} \right\rceil \tag{4.7}$$

Essa simplificação de cenário é coerente, por exemplo, com aplicações paralelas que utilizam a técnica de *checkpointing* para tolerância à faltas (SATO et al., 2012).

Nessa proposta está sendo considerado que os nodos de armazenamento são homogêneos, ou seja, que apresentam as mesmas características de software e hardware. Essa consideração está de acordo com diversas arquiteturas de sistemas de arquivos paralelos reais (SAINI et al., 2012; CARNS et al., 2009; YU; VETTER; ORAL, 2008). Dessa forma, assume-se que os m nodos de armazenamento apresentam um espaço na memória principal, de tamanho *cache*, destinado a cache de escrita de dados; um dispositivo de armazenamento, com taxa de transferência B_{disp} e latência L_{disp}; e um tempo de serviço t_{serv}.

O tamanho *cache* delimita a quantidade de dados $d_a(j)$ que o nodo de armazenamento a_j ($j \in [1, m]$) pode manter em memória sem que a sincronização com o dispositivo de armazenamento seja realizada. As características do dispositivo de armazenamento B_{disp} e L_{disp} correspondem, respectivamente, a capacidade de escrita em bits por unidade de tempo e a custos próprios do dispositivo relacionados ao processo de escrita. Para discos rígidos, L_{disp} representa atrasos referentes às movimentações das cabeças de leitura e escrita, por exemplo. O tempo t_{serv} se refere a custos de computação, interrupção e outros, inerentes ao atendimento de requisições nos nodos de armazenamento.

A quantidade de requisições N_{ra} que cada um dos m nodos de armazenamento precisará servir depende da quantidade de requisições N_{rc} enviadas pelos n nodos de computação e da quantidade de nodos

de armazenamento, conforme definido na Equação (4.8).

$$N_{ra} = \left\lceil \frac{N_{rc}}{m} \right\rceil \tag{4.8}$$

De maneira similar, a quantidade de faixas N_{sa} que cada nodo de armazenamento mantém depende da quantidade de faixas N_{sc} enviadas pelos n nodos de computação e da quantidade de nodos de armazenamento, conforme definido na Equação (4.9).

$$N_{sa} = \left\lceil \frac{N_{sc}}{m} \right\rceil \tag{4.9}$$

Como visto no capítulo anterior, a representação da rede de interconexão é um dos principais desafios da modelagem de sistemas de arquivos paralelos. Uma rede de interconexão pode ser caracterizada pela sua topologia, taxa de transmissão nominal (no inglês, *bandwidth*) e latência (PETERSON; DAVIE, 2003). Esse trabalho considera redes de interconexão nas quais os nodos de computação e de armazenamento estão interligados por meio de um switch de rede e se comunicam diretamente. Considera-se também que a taxa de transmissão nominal no lado dos nodos de computação, B_{netC}, possa ser diferente da taxa de transmissão nominal no lado dos nodos de armazenamento, B_{netA}. A latência, enquanto tempo de propagação, não foi considerada no modelo proposto neste trabalho por uma questão de simplificação. Seu impacto em redes cujas distâncias são relativamente pequenas (atrasos de algumas dezenas de microsegundos), como usualmente ocorre em arquiteturas de sistemas de arquivos paralelos, pode ser considerado desprezível (SAINI et al., 2012; CARNS et al., 2009; YU; VETTER; ORAL, 2008).

Para representar o tempo gasto com a transmissão dos dados na rede de interconexão em um cenário de uso, esse modelo estima a taxa de transmissão efetiva (no inglês, *throughput*) da rede de interconexão, T_{net}. Essa taxa reflete a quantidade de bits efetivamente transmitidos pela rede por unidade de tempo. Nesse trabalho, se adotou para o cálculo da taxa de transmissão efetiva, o *fator médio de recepção* σ proposto por Pinkston e Duato (2006), cuja fórmula para redes interligadas por switch é dada pela Equação (4.10).

$$\sigma = \log_2(X)^{(-\frac{1}{4})} \tag{4.10}$$

onde X é a quantidade de interfaces de rede se comunicando. No con-

texto desse trabalho, $X = n + m$, ou seja, a soma do número de nodos de computação e de armazenamento.

A taxa de transmissão efetiva T_{net} de um cenário de uso composto por n nodos de computação e m nodos de armazenamento é proporcional ao fator médio de recepção σ e limitada pela menor taxa nominal agregada entre os lados dos nodos de computação e de armazenamento, conforme definido pelas Equações (4.11), (4.12) e (4.13).

$$B_{netAggC} = n \times B_{netC} \tag{4.11}$$

$$B_{netAggA} = m \times B_{netA} \tag{4.12}$$

$$T_{net} = \sigma \times \min\{B_{netAggC}, B_{netAggA}\} \tag{4.13}$$

Vale ressaltar que o fator σ atua nesse modelo como uma simplificação para efeitos de concorrência de acesso, como contenção e congestionamento, reduzindo a taxa de transmissão efetiva na presença de uma maior quantidade de nodos.

O modelo proposto nesse trabalho permite estimar duas métricas de desempenho do armazenamento em sistemas de arquivos paralelos: vazão e tempo de resposta. A vazão T do sistema de arquivos paralelo em um cenário de uso foi definida como a capacidade de armazenamento do sistema em bits por unidade de tempo. Seu valor é diretamente proporcional à quantidade total de dados enviados para o sistema e inversamente proporcional ao tempo necessário para concluir todas as operações definidas pelo cenário de uso, o que, no contexto desse trabalho, foi chamado de tempo de resposta. O valor da vazão é estimado pela Equação (4.14).

$$T = \frac{\sum_{i=1}^{n} d_c(i)}{t} \tag{4.14}$$

Considerando a simplificação de balanceamento adotada nessa proposta, a Equação (4.14) pode ser simplificada, resultando na Equação (4.15).

$$T = \frac{n \times d_c}{t} \tag{4.15}$$

A aproximação proposta nesse trabalho para o tempo de resposta do armazenamento em um sistema de arquivos paralelo para um determinado cenário de uso considera três casos:

Caso 1: os dados são armazenados apenas na cache dos nodos de armazenamento, ou seja, $d_a \leq cache$.

Caso 2: a quantidade de dados ultrapassa o limite da cache e a sin-

cronização com o dispositivo de armazenamento é disparada, ou seja, $d_a > cache$.

Caso 3: ao final da operação, os dados em cache são sincronizados com o dispositivo de armazenamento por solicitação dos clientes (chamada de sistema `fsync`).

Vale ressaltar que essa diferenciação na aproximação do tempo de resposta é uma contribuição original dessa pesquisa, cujos resultados preliminares para os casos 1 e 2 foram publicados em (INACIO et al., 2015).

Quando a quantidade de dados d_a recebida em cada nodo de armazenamento é menor que o tamanho da cache (caso 1), o tempo de resposta é aproximado pelo tempo necessário para que os dados sejam transmitidos pela rede de interconexão, considerando uma taxa efetiva de T_{net} bits por unidade de tempo, mais os custos de atendimento das N_{ra} requisições nos nodos de armazenamento, conforme indicado pela Equação (4.16).

$$t_{caso1} = \frac{n \times d_c}{T_{net}} + N_{ra} \times t_{serv} \tag{4.16}$$

No caso 2, a quantidade de dados d_a recebida pelos nodos de armazenamento é superior ao tamanho reservado para cache. Quando isso ocorre, inicia-se um processo de sincronização de dados da cache para o dispositivo de armazenamento. A aproximação para o tempo de resposta nesse caso considera: o tempo necessário para que as caches dos m nodos de armazenamento sejam preenchidas, considerando uma rede de interconexão com taxa efetiva T_{net}; o tempo para que a quantidade de requisições necessárias para completar a cache sejam servidas por um nodo de armazenamento; e o tempo necessário para que a quantidade de dados que excedeu a cache seja armazenada no dispositivo de armazenamento, considerando uma taxa de transferência B_{disp} e um atraso proporcional a latência L_{disp} do dispositivo e ao número de faixas de dados que serão sincronizados, conforme indicam as Equações (4.17).

$$\begin{aligned} t_1 &= \frac{m \times cache}{T_{net}} \\ t_2 &= \left\lceil \frac{cache}{r} \right\rceil \times t_{serv} \\ t_3 &= \frac{d_a - cache}{B_{disp}} + \left(N_{sa} - \left\lfloor \frac{cache}{s} \right\rfloor \right) \times L_{disp} \\ t_{caso2} &= t_1 + t_2 + t_3 \end{aligned} \tag{4.17}$$

O terceiro caso aborda cenários em que os processos executando nos n nodos de computação solicitam explicitamente, por meio da chamada de sistema `fsync`, a sincronização dos dados enviados no dispositivo de armazenamento. Essa é uma prática adotada por algumas aplicações para garantir que na finalização da operação de escrita os dados estejam persistidos no dispositivo de armazenamento e não apenas na memória principal. Algumas condições são consideradas na aproximação do tempo para esse caso. O tempo consumido na transmissão dos dados na rede de interconexão vai ser no máximo o tempo necessário para preencher o espaço da cache, visto que nessa situação, a sincronização dos dados é iniciada independente da requisição do cliente. Da mesma forma, os custos associados ao atendimento das requisições vão ser limitados pelo número de requisições necessários para preencher a cache. Diferentemente do caso 2, no terceiro caso todos os dados enviados são sincronizados no dispositivo de armazenamento, conforme indicam as Equações (4.18).

$$\begin{aligned} t_1 &= \min\left\{\left(\frac{n \times d_c}{T_{net}}\right), \left(\frac{m \times cache}{T_{net}}\right)\right\} \\ t_2 &= \min\left\{N_{ra}, \left\lceil\frac{cache}{r}\right\rceil\right\} \times t_{serv} \\ t_3 &= \frac{d_a}{B_{disp}} + N_{sa} \times L_{disp} \\ t_{caso3} &= t_1 + t_2 + t_3 \end{aligned} \tag{4.18}$$

Em resumo, o tempo de resposta do sistema de arquivos paralelo pode ser aproximado pela Equação (4.19).

$$t = \begin{cases} t_{caso1} & , d_a \leq cache \wedge \neg fsync \\ t_{caso2} & , d_a > cache \wedge \neg fsync \\ t_{caso3} & , fsync \end{cases} \tag{4.19}$$

4.3 CONSIDERAÇÕES

Nesse capítulo, a formalização do modelo analítico multivariado proposto nessa pesquisa é apresentada. O objetivo desse modelo é representar o comportamento do desempenho do armazenamento em sistemas de arquivos paralelos baseado em um conjunto de variáveis gerais. Um dos principais desafios da representação de modelos multivari-

ados está justamente na identificação do relacionamento entre diversas variáveis independentes e a variável dependente.

A construção do modelo apresentado nesse capítulo se deu de forma iterativa e por meio de diferentes abordagens. Primeiramente, foi realizado um estudo das características de diferentes sistemas de arquivos paralelos e de propostas de modelagem anteriores encontradas na literatura. Esse estudo permitiu a seleção de um conjunto inicial de variáveis e a formalização de algumas expressões básicas. Um amplo estudo experimental, considerando diferentes plataformas computacionais e casos de uso, possibilitou a caracterização do desempenho do sistema, que por sua vez, permitiu o refinamento do conjunto de variáveis e a formalização de comportamentos mais complexos.

Vale ressaltar que a proposta dessa pesquisa se diferencia das anteriores por alguns aspectos. Essa proposta avança no sentido de representar o impacto de um grande número de variáveis no desempenho do armazenamento em sistemas de arquivos paralelos de forma analítica, para um espectro maior de cenários de uso. Se diferencia também por ter sido construída e avaliada a partir de um número bastante significativo de experimentos conduzidos em ambientes reais. O detalhamento dos experimentos e os resultados da caracterização e avaliação do modelo proposto são apresentados no próximo capítulo.

5 AMBIENTE E RESULTADOS EXPERIMENTAIS

Para compreender o comportamento real do desempenho de sistemas de arquivos paralelos e avaliar o modelo proposto nesse trabalho, um amplo conjunto de experimentos foi realizado. Nesses experimentos foram avaliados o tempo de resposta e a vazão do sistema de arquivos paralelo para diferentes cenários de uso, que, no contexto dessa pesquisa, representam combinações de valores para o conjunto de fatores considerados (ver Seção 4.1). No total, foram 1.111.346 observações formalmente coletadas durante os experimentos.

A Seção 5.1 descreve as quatro plataformas computacionais utilizadas e o método experimental. A caracterização do efeito de diferentes variáveis no desempenho do armazenamento em sistemas de arquivos paralelos é apresentada na Seção 5.2. Na Seção 5.3, se discute a avaliação do modelo proposto nessa pesquisa.

5.1 AMBIENTE EXPERIMENTAL E MÉTODO

Para a realização dos experimentos conduzidos nessa pesquisa, quatro ambientes computacionais foram utilizados: dois *clusters* dedicados, Sagittaire e Graphene, no Grid'5000 (CAPPELLO et al., 2005), na França; um ambiente na nuvem pública da Microsoft Azure (MICROSOFT, 2014); e um ambiente virtualizado no Laboratório de Pesquisa em Sistemas Distribuídos (LAPESD) da Universidade Federal de Santa Catarina (UFSC).

O *cluster* Grid'5000 Sagittaire conta com 79 nodos Sun Fire V20z. Cada nodo possui um processador AMD Opteron 250 com 2 núcleos a 2,4 GHz, 2 GiB de memória RAM e um disco SCSI de 73 GiB. Os nodos estão interconectados por meio de uma rede Gigabit Ethernet. Em cada nodo foi executado o sistema operacional Debian 6, com *kernel* 2.6.32 e sistema de arquivos local ext3. Nesse ambiente, 32 nodos foram utilizados para armazenamento e 16 para computação.

O *cluster* Grid'5000 Graphene apresenta 144 nodos Carri System CS-5393B interligados por um rede Infiniband-20G. Cada nodo conta com um processador Intel Xeon X3440 com 4 núcleos a 2,53 GHz, 16 GiB de memória RAM e um disco SATA II de 320 GiB. O Debian 6 foi o sistema operacional utilizado nos nodos, com *kernel* 2.6.32 e sistema de arquivos local ext3. 32 nodos foram dedicados à função de armazenamento e 16 a função de computação.

Um ambiente na nuvem pública da Microsoft Azure foi criado utilizando 32 máquinas virtuais (MV) do tipo *Standard Small* (A1). Essas instâncias são compostas de um processador virtual, 1,75 GiB de memória RAM e um disco virtual de 30 GiB. Em cada MV foi instalado o sistema operacional CentOS 6.5 (*kernel* 2.6.32), com sistema de arquivos local ext4. Das 32 MVs disponíveis, 16 foram utilizadas como nodos de armazenamento e 16 como nodos de computação.

Um ambiente virtualizado foi criado no LAPESD utilizando uma máquina IBM System x3650 M3 com 2 processadores Intel Xeon E5649 a 2,53 GHz, 32 GiB de memória RAM e 4 discos SAS organizados em RAID-5. Cada processador apresenta 6 núcleos *hyper threading* (12 *threads*). O VMWare ESXi 5.5 foi utilizado como *hypervisor* para criação de 24 MVs. Um processador virtual, 1 GiB de memória RAM e um disco virtual de 20 GiB foi alocado para cada MV. O sistema operacional CentOS 6.5, com *kernel* 2.6.32 e sistema de arquivos local ext4 foi instalado em cada MV. 16 das 24 MVs foram utilizados como nodos de armazenamento e 8 como nodos de computação.

O sistema de arquivos paralelo OrangeFS versão 2.8.8 (ORANGEFS, 2014) foi instalado em todos os ambientes computacionais. Esse sistema de arquivos paralelo foi escolhido por ser utilizado em diversos ambientes reais e trabalhos de pesquisa sobre armazenamento paralelo. A geração da carga de trabalho foi realizada utilizando o benchmark *Interleaved Or Random* (IOR) (LAWRENCE LIVERMORE NATIONAL LABORATORY, 2014). O IOR é um benchmark desenvolvido e amplamente utilizado para avaliação de sistemas de E/S de alto desempenho. Suas facilidades de configuração permitem geração diferentes tipos de padrões de acesso e cargas de trabalho.

Cada cenário de uso, caracterizado por uma combinação de valores para os diferentes fatores considerados, foi avaliado por no mínimo 30 repetições (observações experimentais). Em alguns casos, um número maior de repetições foi necessário em função da grande variabilidade observada nas amostras. A Tabela 2 apresenta os valores definidos para diversos fatores nos experimentos dessa pesquisa.

Na Tabela 3, são apresentados valores para fatores inerentes às características dos ambientes experimentais: Grid'5000 Sagittaire (G5K Sag.), Grid'5000 Graphene (G5K Gra.), Microsoft Azure (Azure) e LAPESD. Esses valores foram obtidos utilizando ferramentas como IPERF3 (ESNET, 2014), dstat (WIEERS, 2014) e strace (LEVIN; MCGRATH; AKKERMAN, 2014), além de rastros de execução capturados pelos logs do OrangeFS.

Tabela 2: Valores definidos para análise experimental.

Fator	**Descrição**	**Valores**
n	Número de nodos de computação	1, 2, 4, 8, 16
m	Número de nodos de armazenamento	1, 2, 4, 8, 16, 32
d_c	Quantidade de dados enviada por nodo de computação (em MiB)	1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024
r	Tamanho da requisição de escrita (em KiB)	32, 64, 128, 256, 512, 1024
s	Tamanho da faixa do sistema de arquivos paralelo (em KiB)	16, 32, 64, 128, 256
$fsync$	Invocação explícita de sincronização de dados no dispositivo de armazenamento	sim, não
TCP	Controle de congestionamento TCP	Reno, Westwood, Highspeed, CUBIC
API	Interface com sistema de arquivos paralelo	POSIX, MPI-IO

5.2 CARACTERIZAÇÃO DE DESEMPENHO

Visando prover uma caracterização do comportamento real desses sistemas, uma extensa análise experimental foi realizada verificando o efeito de diferentes fatores no desempenho do armazenamento em sistemas de arquivos paralelos. Esse esforço permitiu não apenas reproduzir efeitos observados em estudos anteriores (KUNKEL; LUDWIG, 2007; CARNS et al., 2009; SAINI et al., 2012) como também identificar novos comportamentos relativos a influência de fatores não explorados anteriormente. Nessa seção, são apresentados apenas os resultados que refletem contribuições originais desse trabalho de pesquisa.

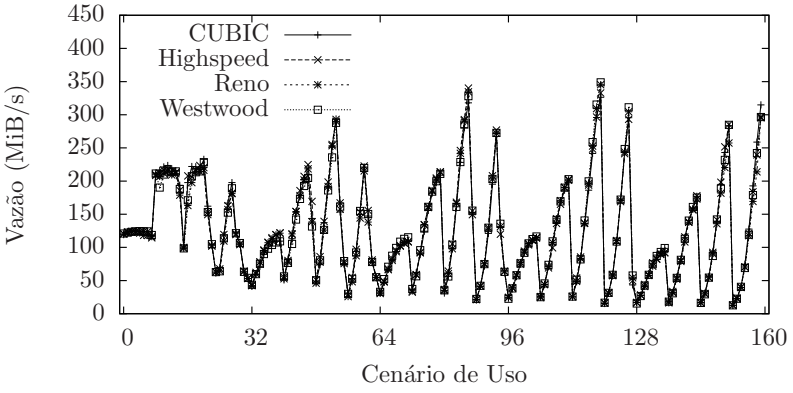
Cenários típicos de uso de sistemas de arquivos paralelos envolvem uma grande quantidade de clientes simultâneos. Isso pode gerar, dependendo da infraestrutura disponível, um alto nível de concorrência pelo acesso a rede de interconexão. Nesse trabalho, o desempenho do sistema de arquivos paralelo foi avaliado para quatro algoritmos de controle de congestionamento TCP: CUBIC, Highspeed, Reno e Westwood.

Tabela 3: Valores de fatores medidos nos ambientes experimentais.

Fator	Descrição	G5K Sag.	G5K Gra.	Azure	LAPESD
B_{netC}	Taxa de transmissão nominal da rede nos nodos de computação (em Gbps)	1	20	1	1
B_{netA}	Taxa de transmissão nominal da rede nos nodos de armazenamento (em Gbps)	1	20	1	4
B_{disp}	Taxa de transferência do dispositivo de armazenamento (em MiB/s)	60	120	50	50
L_{disp}	Atraso decorrente de características do dispositivo de armazenamento (em segundos)	0,0002	0,0002	0,0002	0,0004
cache	Tamanho da cache de escrita (em MiB)	200	1600	175	100
t_{serv}	Custo de atendimento de uma requisição (em segundos)	0,0008	0,0002	0,0005	0,0001

A Figura 8 ilustra o comportamento observado para 160 cenários de uso no ambiente LAPESD. Os cenários crescem em complexidade da

Figura 8: Comparação da vazão do sistema de arquivos paralelos com diferentes algoritmos de controle de congestionamento TCP no ambiente LAPESD.

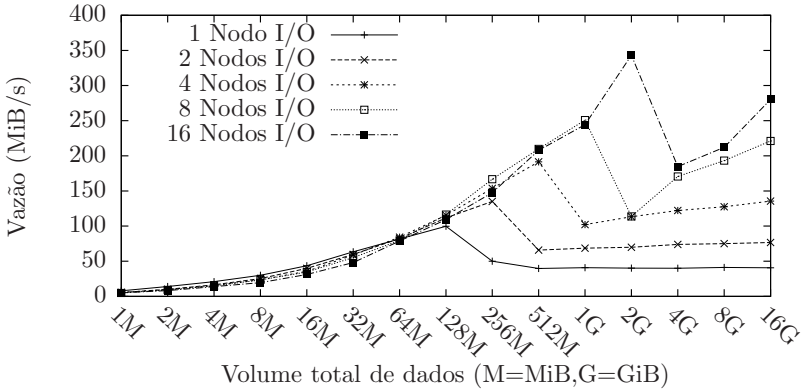


esquerda para a direita, ou seja, o primeiro cenário corresponde a um caso com um nodo de armazenamento e um nodo de computação, escrevendo arquivos de 1 MiB; e o último caso corresponde a 16 nodos de armazenamento e 8 nodos de computação, escrevendo arquivos de 128 MiB.

Observa-se no gráfico a impossibilidade de distinguir o comportamento dos diferentes algoritmos. De fato, a média das diferenças de desempenho entre o melhor e o pior algoritmo para cada cenário de uso foi de aproximadamente 6%. Sendo assim, conclui-se que para os cenários de uso explorados, a opção por um dos quatro algoritmos de controle de congestionamento avaliados não contribui para o desempenho do sistema de arquivos paralelo.

Como mencionado anteriormente, a representação do efeito da cache de escrita dos nodos de armazenamento no desempenho do sistema de arquivos paralelo é um contribuição original dessa pesquisa. Esse efeito foi identificado pela análise do desempenho do armazenamento de acordo com o volume total de dados enviados para o sistema. O volume total de dados corresponde ao somatório da quantidade de dados enviado por cada cliente. As Figuras 9, 10 e 11 ilustram esse efeito para os ambientes Microsoft Azure, LAPESD e Grid'5000 Sagittaire, respectivamente. O tamanho da faixa utilizada no sistema de arquivos paralelo para esse cenário foi de 64 KiB e cada cliente utilizando requisições de 1 MiB.

Figura 9: Efeito da cache de escrita no desempenho do sistema de arquivos paralelos no ambiente Microsoft Azure.



É possível observar nos resultados que a vazão do sistema cresce

Figura 10: Efeito da cache de escrita no desempenho do sistema de arquivos paralelos no ambiente LAPESD.

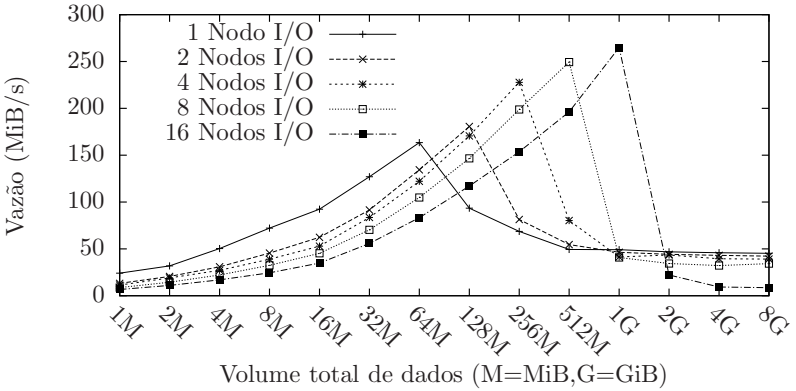
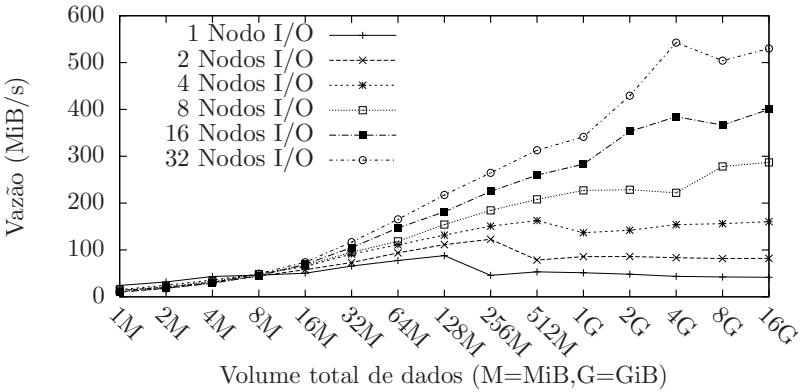


Figura 11: Efeito da cache de escrita no desempenho do sistema de arquivos paralelos no cluster Grid'5000 Sagittaire.

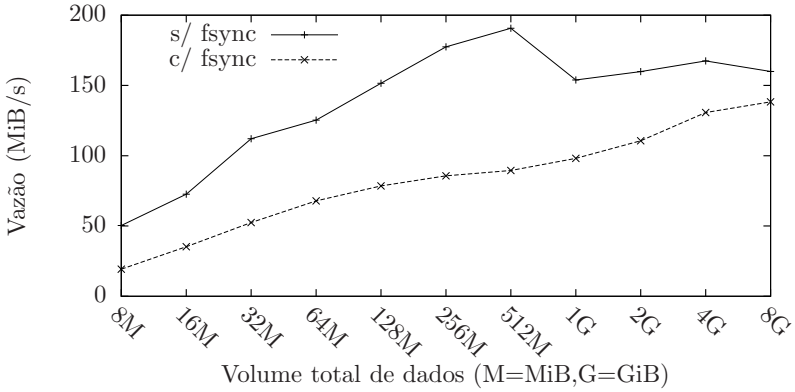


com o volume de dados, atinge um ponto máximo e descresce. Essa queda na vazão reflete a sincronização dos dados no dispositivo de armazenamento secundário em função do preenchimento da cache. O volume de dados no qual o ponto máximo é obtido dobra quando o número de nodos de armazenamento também dobra. Isso ocorre porque com o aumento do número de nodos de armazenamento, cada nodo recebe uma parcela menor dos dados. Dessa forma, o sistema como um

todo é capaz de suportar a escrita de um volume maior de dados na cache, oferecendo melhor desempenho.

Outro comportamento identificado na caracterização realizada nessa pesquisa é o efeito da invocação explícita da sincronização dos dados da cache de escrita no dispositivo de armazenamento pelos clientes. Essa invocação é realizada por meio da chamada de sistema `fsync`. As Figuras 12, 13 e 14 ilustram a vazão do armazenamento em sistemas de arquivos paralelos com e sem a chamada de `fsync` para diferentes volumes de dados, para os ambientes Grid'5000 Sagittaire, Microsoft Azure e LAPESD, respectivamente. Os resultados refletem cenários com 8 nodos de computação, 4 nodos de armazenamento, tamanhos de requisição de 1 MiB e tamanho de faixa de 64 KiB.

Figura 12: Desempenho do sistema de arquivos paralelos no cluster Grid'5000 Sagittaire com e sem invocação da chamada de sistema `fsync`.



Dois aspectos são particularmente interessantes nesses resultados. Primeiro, o significativo ganho de desempenho proporcionado pelo uso da cache de escrita. Uma vazão superior foi observada na curva que representa o comportamento sem a chamada de `fsync`. Segundo, a aproximação das duas curvas quando o volume de dados é superior ao espaço de cache provido pelo sistema. São cenários em que, mesmo sem a invocação do `fsync`, os dados são sincronizados com o dispositivo de armazenamento.

Figura 13: Desempenho do sistema de arquivos paralelos no ambiente Microsoft Azure com e sem invocação da chamada de sistema `fsync`.

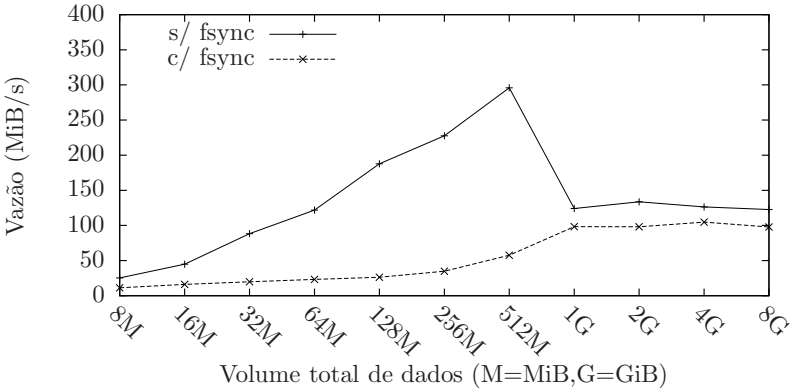
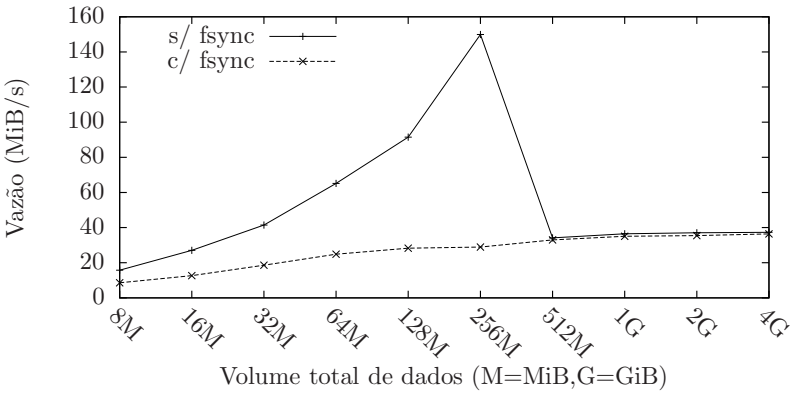


Figura 14: Desempenho do sistema de arquivos paralelos no ambiente LAPESD com e sem invocação da chamada de sistema `fsync`.



5.3 AVALIAÇÃO DO MODELO

Para avaliar o modelo proposto nesse trabalho (ver Seção 4.2), os valores estimados pelo modelo foram comparados com os valores medidos nos quatro ambientes experimentais apresentados na Seção 5.1 para diferentes cenários de uso. Em função do grande volume de resultados, um subconjunto desses, considerados mais relevantes para

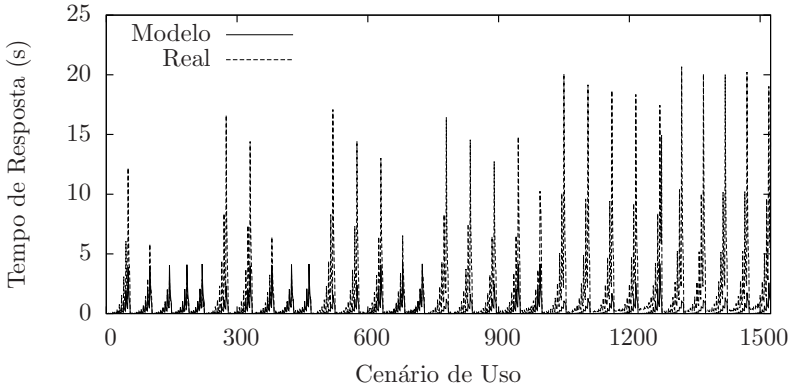
a argumentação dessa avaliação, são apresentados.

Os resultados apresentados referem-se aos tempos de resposta estimados e medidos de acordo com diferentes cenários de uso. A complexidade dos cenários de uso cresce da esquerda para a direita no eixo x dos gráficos apresentados nessa seção. Esse crescimento se dá pelo aumento dos valores das variáveis (ver Tabela 2) na seguinte ordem: tamanho da faixa, tamanho da requisição, quantidade de dados, número de nodos de computação e número de nodos de armazenamento.

Por exemplo, um primeiro cenário seria definido por um nodo de armazenamento, um nodo de computação, escrevendo 1 MiB de dados, utilizando requisições de 32 KiB, com tamanho de faixa igual a 16 KiB. O cenário seguinte manteria o valor de todas as variáveis, com exceção do tamanho de faixa, que aumenteria para 32 KiB. O processo continua até que se tenha no último cenário, mais a direita no gráfico, a configuração máxima para aquele ambiente, em termos de quantidade de nodos de computação e de armazenamento.

Vale observar que nem todos os cenários (combinações) possíveis foram medidos para todos os ambientes por limitações de tempo e de acesso aos ambientes computacionais para a realização dos experimentos. Entretanto, comparado ao que foi observado em trabalhos anteriores, o número de cenários avaliados nessa pesquisa é significativamente superior e pode ser considerado suficiente para as conclusões obtidas.

Figura 15: Comparação do desempenho estimado pelo modelo e medido no cluster Grid'5000 Graphene para cenários no caso 1.

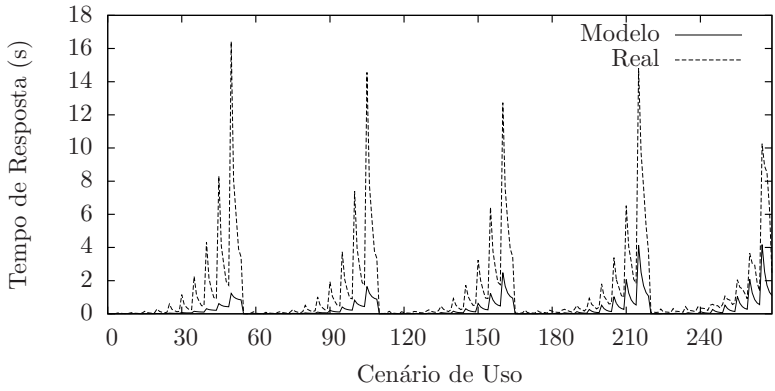


A avaliação do modelo proposto foi realizada de acordo com os

três casos representados (ver Seção 4.2). Para demonstrar a avaliação do caso 1, em que os dados são armazenados apenas na cache dos nodos de armazenamento, os resultados obtidos no cluster Grid'5000 Graphene foram utilizados. A Figura 15 ilustra a comparação entre o desempenho estimado pelo modelo e medido experimentalmente. Por meio desse panorama geral, é possível observar que, em termos comportamentais, não há grandes desvios entre o desempenho medido e estimado. Em termos absolutos, por outro lado, obsera-se que o tempo de resposta medido é, em grande parte dos casos, superior ao estimado.

Na Figura 16, um subconjunto dos cenários, em que o número de nodos de armazenamento foi limitado em 8, é utilizado para ilustrar com mais detalhes esse comportamento. Nesse gráfico, as caracterís-

Figura 16: Efeito da taxa de transissão da rede de interconexão nas estimativas do modelo para o cluster Grid'5000 Graphene nos cenários do caso 1.

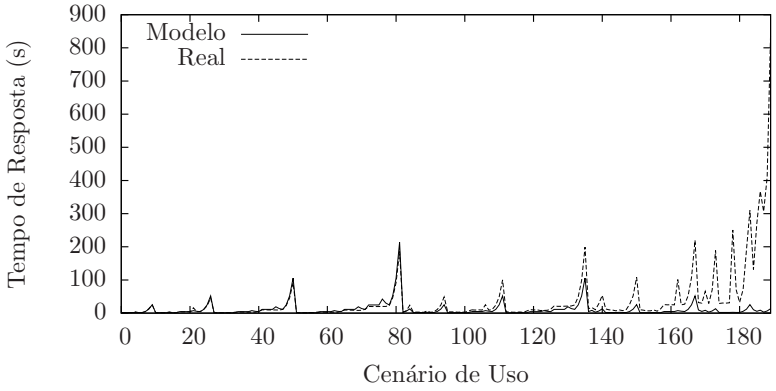


ticas qualitativas e quantitativas das estimativas para esse ambiente ficam mais evidentes. Observa-se a reprodução correta do comportamento, porém com valores subestimados. Por meio de uma análise experimental mais pontual, verificou-se que a causa dessa diferença quantitativa pode estar relacionada com a taxa de transmissão da rede de interconexão. No modelo, utilizou-se como parâmetro a taxa nominal (20 Gbps) divulgada no site do Grid'5000 (GRID'5000, 2014). Experimentos com o benchmark IPERF3, no entanto, apresentaram uma taxa máxima de 14 Gbps. Considerando uma taxa de interconexão menor, o tempo de resposta estimado pelo modelo aumenta, se aproximando dos valores medidos. É interessante observar, com base nesse incidente,

como o modelo proposto nesse trabalho pode auxiliar na identificação de comportamentos não esperados em um ambiente de sistemas de arquivos paralelo.

Os resultados experimentais medidos no ambiente LAPESD foram utilizados para demonstrar a capacidade de representação do modelo proposto para o caso 2. No caso 2, a quantidade de dados enviada para o sistema ultrapassa o limite da cache e a sincronização com o dispositivo de armazenamento é disparada. A Figura 17 apresenta a comparação entre os tempos de resposta medidos experimentalmente e os estimados pelo modelo para diferentes cenários de uso.

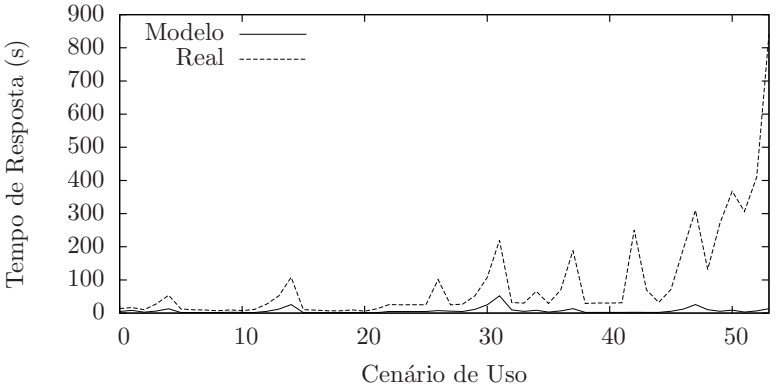
Figura 17: Comparação do desempenho estimado pelo modelo e medido no ambiente LAPESD para cenários no caso 2.



É possível observar nesses resultados que o modelo foi capaz de representar com sucesso tanto o comportamento quanto a amplitude do desempenho para pouco mais da metade dos cenários de uso nesse ambiente. Entretanto, à medida que a quantidade de nodos de armazenamento aumenta, percebe-se, primeiramente, um desvio na amplitude do desempenho, seguido por desvios na representação do próprio comportamento do sistema.

Para detalhar esse efeito, os resultados de um subconjunto desses cenários de uso, com 4, 8 e 16 nodos de armazenamento, é apresentado na Figura 18. Nesse gráfico, os desvios na comparação podem ser observados mais claramente. A causa para o crescimento no desvio acompanhar o crescimento no número de nodos de armazenamento pode estar relacionada à virtualização. Esse ambiente foi criado sobre uma única máquina física. Dessa forma, todas as máquinas virtuais

Figura 18: Efeito da virtualização nas estimativas do modelo para o ambiente LAPESD nos cenários do caso 2.

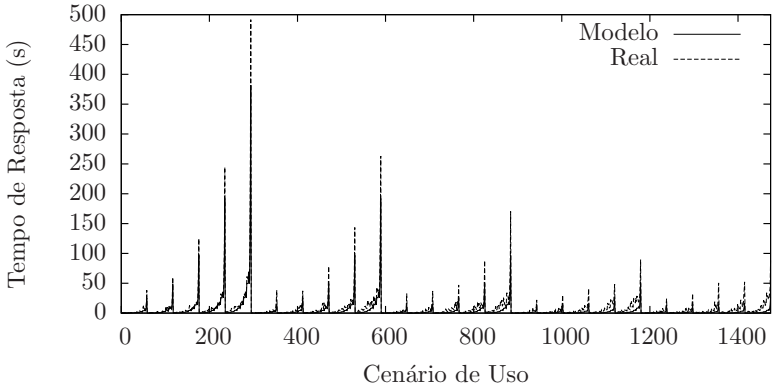


(MV) disputam os recursos disponíveis. No caso 2, quando a sincronização dos dados é disparada, as MVs competem pelo acesso aos discos rígidos. Isso provoca um atraso maior no processo, não representado no modelo. Esse efeito foi verificado, por meio da execução de testes simultâneos de escrita local em cada um dos nodos de armazenamento. Vale ressaltar que, apesar dos desvios serem consideráveis, essa arquitetura não é uma opção comum em ambientes reais de armazenamento paralelo e, portanto, não desqualifica significativamente a capacidade de representação do modelo.

A demonstração da avaliação do caso 3, em que a sincronização dos dados é invocada explícitamente pelos clientes, do modelo proposto é realizada utilizando os resultados do ambiente Microsoft Azure. A Figura 19 apresenta a comparação do desempenho estimado pelo modelo e medido no ambiente. Apesar de tratar-se também de um ambiente virtualizado, o modelo foi capaz de reproduzir qualitativamente o comportamento do desempenho para os diferentes cenários de uso no ambiente criado na nuvem pública Microsoft Azure. Diferentemente da arquitetura do ambiente LAPESD, a configuração realizada na Microsoft Azure pode efetivamente vir a ser utilizada no futuro, considerando esforços existentes para execução de computação de alto desempenho sobre plataformas de nuvem computacional. Do ponto de vista quantitativo, observa-se que o modelo subestima o tempo de resposta para a maior parte dos cenários.

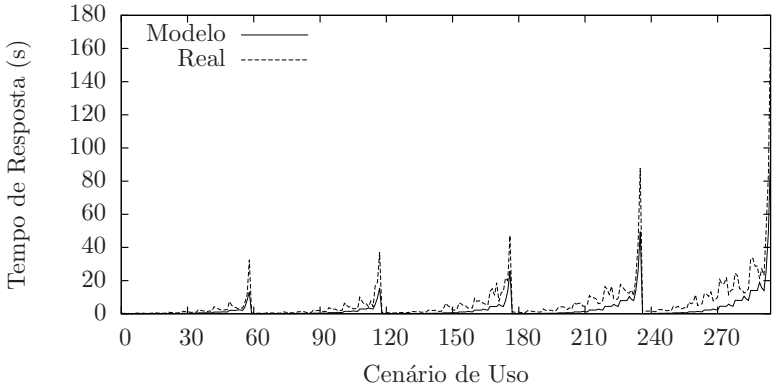
Para explorar esse comportamento, os resultados de um subcon-

Figura 19: Comparação do desempenho estimado pelo modelo e medido no ambiente Microsoft Azure para cenários no caso 3.



junto dos cenários de uso, em que o número de nodos de armazenamento foi fixado em 4, são apresentados na Figura 20. Nesse ambiente, assim

Figura 20: Detalhe dos desvios nas estimativas do modelo para o ambiente Microsoft Azure nos cenários do caso 3.

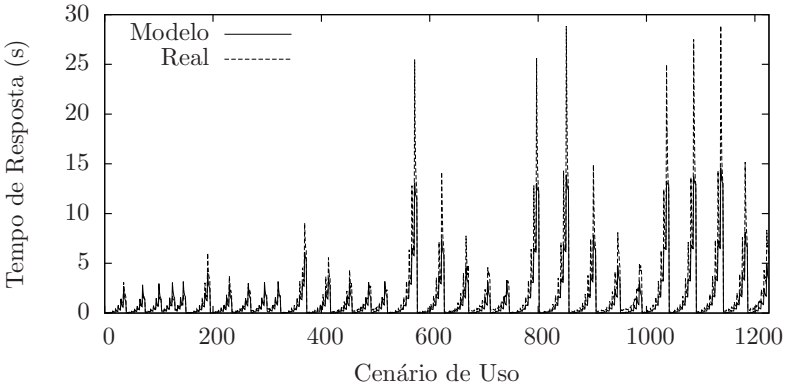


como no do LAPESD, é esperado que um desvio entre o valor estimado e o valor medido para o desempenho exista em função do compartilhamento de recursos físicos entre MVs. Para esse ambiente, no entanto, as informações disponíveis sobre a alocação das MVs não são suficientes para determinar as causas do comportamento observado. Além da

concorrência pelos recursos físicos, variações de até 300 Mbps nas taxas de transmissão da rede entre MVs, medidas utilizando o benchmark IPERF3, podem ter provocado os desvios observados.

Os melhores resultados na comparação de valores medidos e estimados foram obtidos para o cluster Grid'5000 Sagittaire. As Figuras 21, 22 e 23 demonstram esses resultados para os três casos abordados. Se observa nesses gráficos a similaridade entre o comportamento representado pelo modelo e o desempenho medido experimentalmente.

Figura 21: Comparação do desempenho estimado pelo modelo e medido no cluster Grid'5000 Sagittaire para cenários no caso 1.



Esses melhores resultados podem ser atribuídos a alguns aspectos. Diferentemente dos ambientes virtualizados, como os ambientes LAPESD e Microsoft Azure, condições de compartilhamento de recursos não foram observados no cluster. De fato, cada nodo do cluster possui um conjunto próprio de processores, memória e discos rígidos.

Com relação ao cluster Grid'5000 Graphene, a diferença nos resultados pode estar vinculada a ocupação do cluster. O Graphene é o cluster com a maior quantidade de nodos no Grid'5000 e sua utilização é altamente concorrida. Durante os experimentos, foram comuns os casos em que foi possível reservar apenas uma parte dos nodos disponíveis. Logo, enquanto os experimentos dessa pesquisa foram desenvolvidos, diversos outros experimentos estavam possivelmente em execução ao mesmo no cluster, consumindo recursos de rede, por exemplo. Essa condição praticamente não ocorreu nas reservas do cluster Grid'5000 Sagittaire, em que a maior parte do cluster era dedicada aos experi-

Figura 22: Comparação do desempenho estimado pelo modelo e medido no cluster Grid'5000 Sagittaire para cenários no caso 2.

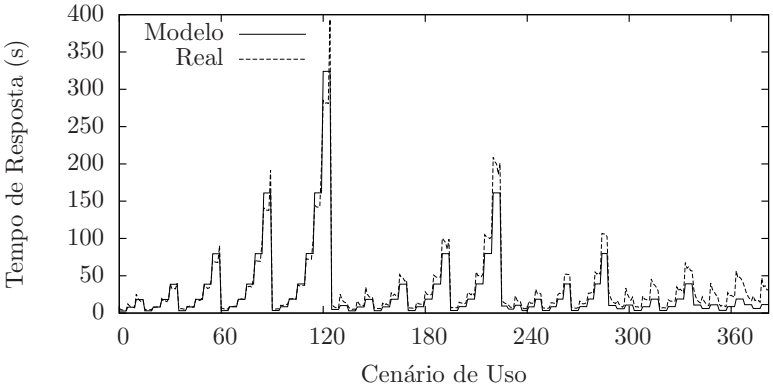
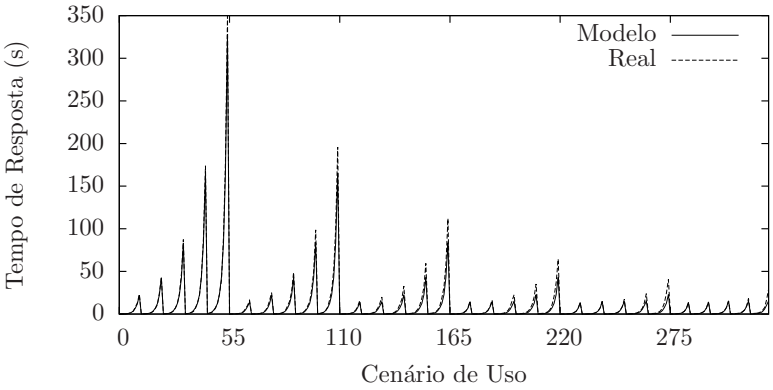


Figura 23: Comparação do desempenho estimado pelo modelo e medido no cluster Grid'5000 Sagittaire para cenários no caso 3.



mentos dessa pesquisa.

Concluindo essa seção, a avaliação da acurácia do modelo proposto, em termos quantitativos, para os três casos de teste e os quatro ambientes experimentais é apresentada na Tabela 4. O erro percentual médio absoluto (do inglês, *mean absolute percentage error* - MAPE),

Tabela 4: Erro percentual médio absoluto do modelo para os quatro ambientes experimentais.

Ambientes	**Caso 1**	**Caso 2**	**Caso 3**	**Total**
LAPESD	54,08%	63,44%	67,11%	60,41%
Microsoft Azure	66,25%	48,14%	68,95%	65,29%
Grid'5000 Sagittaire	43,36%	32,60%	34,53%	39,75%
Grid'5000 Graphene	74,62%	144,13%	-	78,86%

foi utilizado para essa avaliação e é dado pela Equação (5.1).

$$MAPE = \frac{1}{n} \times \sum_{i=1}^{n} \left| \frac{R_i - E_i}{R_i} \right| \times 100\% \tag{5.1}$$

em que n é o número de observações, R_i é o valor medido e E_i é o valor estimado na i-ésima amostra. A acurácia do modelo para o caso 3 no cluster Grid'5000 Graphene não foi avaliada porque experimentos com chamadas de sistemas `fsync` não foram realizados naquele ambiente.

6 CONCLUSÕES E TRABALHOS FUTUROS

Sistemas de arquivos paralelos vêm sendo amplamente utilizados para suportar aplicações que manipulam enormes quantidades de dados, problema esse conhecido como *Big Data*. Contudo, a utilização eficiente dessa solução de armazenamento depende da compreensão do seu comportamento diante de diferentes condições de uso. Essa é uma tarefa complexa, visto o caráter multivariado do problema. O desempenho do sistema de arquivos paralelo depende do relacionamento entre um grande número de variáveis, como o número de nodos de computação e de armazenamento, taxa de transmissão da rede de interconexão, entre outros.

Objetivando prover um melhor entendimento sobre o comportamento do desempenho do armazenamento em sistemas de arquivos paralelos para diferentes configurações e cargas de trabalho, foi proposto nessa pesquisa um modelo analítico multivariado. A opção por um modelo analítico permite que estimativas aproximadas sejam obtidas em um curto período de tempo e com baixo custo computacional.

Um extenso conjunto de experimentos, composto por 1.111.346 observações, combinando diferentes valores para um conjunto de 11 variáveis, foi executado utilizando quatro ambientes computacionais reais: dois clusters dedicados, Sagittaire e Graphene, na Grid'5000 (na França); um ambiente na nuvem pública da Microsoft Azure; e um ambiente virtualizado no LAPESD. Com base nos resultados obtidos, uma caracterização do efeito dos diferentes fatores no desempenho do armazenamento foi realizada. Entre os efeitos observados, a caracterização de três deles são contribuições diferenciais dessa pesquisa.

Verificou-se que, para os cenários de uso analisados, o uso de quatro algoritmos de controle de congestionamento TCP diferentes (CUBIC, Highspeed, Reno, Westwood) não contribuiu significativamente para o desempenho do sistema de arquivos paralelo. A influência da cache de escrita do sistema operacional dos nodos de armazenamento no desempenho geral do sistema de arquivos paralelo foi outro aspecto caracterizado nessa pesquisa. Ganhos de desempenho significativos foram observados para cenários de uso em que as caches dos nodos de armazenamento suportam o volume total de dados enviado para o sistema. Vale ressaltar que em propostas anteriores encontradas na literatura, o efeito da cache foi evitado experimentalmente ou utilizado para justificar erros de predição. Esse trabalho apresenta também a caracterização do efeito de invocações explícitas (chamadas de sistema `fsync`)

de sincronização de dados da cache nos dispositivos de armazenamento. Mostrou-se que essa invocação tem efeito similar ao comportamento observado quando o volume de dados enviado para o sistema é superior às caches dos nodos de armazenamento.

A partir dessa caracterização e das definições fundamentais de sistemas de arquivos paralelos, a formalização do modelo analítico multivariado passou por diversas evoluções, culminando na versão apresentada nesse trabalho. Foram considerados no modelo apenas variáveis presentes na arquitetura básica de sistemas de arquivos paralelos. Ainda assim, esse modelo é capaz de representar relacionamentos complexos, explorando características, como o tamanho da faixa do sistema de arquivos paralelo, e efeitos, como a influência da cache no desempenho, não explorados em trabalhos anteriores.

O modelo foi avaliado com relação à reprodução do comportamento e dos valores estimados, comparando seus resultados com as observações obtidas nos ambientes experimentais. Em termos de comportamento, pode-se considerar que o modelo proposto nessa pesquisa teve sucesso na representação do desempenho do armazenamento em sistemas de arquivos paralelos. Um diferencial desse modelo, com relação aos trabalhos anteriores, é a representação do tempo de resposta de acordo com três casos distintos: um caso para escrita apenas na cache dos nodos de armazenamento; um caso para quando a quantidade de dados é maior que o tamanho da cache e a sincronização com o dispositivo de armazenamento é disparada; e um caso para quando os clientes invocam explicitamente a sincronização com os dispositivos, por meio da chamada de sistema `fsync`.

Com relação a acurácia do modelo, relativa à propriedade de fidelidade dos valores estimados com os valores medidos nos experimentos, desvios percentualmente significativos foram observados. Entretanto, mostrou-se que parte desses desvios podem ser provenientes de situações particulares observadas nos experimentos, como compartilhamento de recursos físicos, diferença entre parâmetros anunciados e observados na prática, entre outros. Em experimentos conduzidos no cluster Grid'5000 Sagittaire, em que essas condições foram menos propícias, um erro percentual médio de 39,75% foi obtido. Comparando esse resultado a resultados obtidos com outros modelos analíticos e baseados em simulação apresentados nessa pesquisa, e considerando a quantidade superior de cenários e observações utilizadas nessa avaliação, pode-se considerar que as estimativas do modelo são apropriadas.

Uma das principais dificuldades encontradas na realização dessa pesquisa foi o tratamento do grande número de variáveis envolvidas. A

explosão combinatorial provocada pela quantidade de variáveis e valores analisados exigiu um grande esforço a nível de realização de experimentos e de análise de resultados. Até mesmo aspectos usualmente considerados simples, como a visualização de resultados, tornam-se particularmente desafiadores para problemas com mais de 4 variáveis. Diversas técnicas estatísticas, de análise multivariada e de aproximação de curvas foram utilizadas no desenvolvimento dessa pesquisa.

Espera-se que o modelo analítico multivariado resultante desse trabalho de pesquisa possa ser utilizado para identificação de comportamentos não esperados em ambientes reais, projetos de novas arquiteturas de armazenamento, dimensionamento de ambientes, suporte para algoritmos de escalonamento e de seleção de recursos, desenvolvimento de simuladores e benchmarks, entre outros.

6.1 TRABALHOS FUTUROS

Embora os resultados obtidos neste trabalho terem sido considerados satisfatórios, o modelo proposto pode ser estendido em trabalhos futuros de diversas maneiras. Entre as possíveis, estão:

- consideração do efeito do compartilhamento de recursos para ambientes de armazenamento em nuvens computacionais;
- representação de operações de leitura;
- aplicação do modelo em soluções de otimização.

REFERÊNCIAS

ARGONNE NATIONAL LABORATORY. **Parallel I/O Benchmarking Consortium**. Mai 2002. Disponível em: <http://www.mcs.anl.gov/research/projects/pio-benchmark>.

BARRO, J. et al. Performance modeling and evaluation of MPI-I/O on a cluster. **Journal of Information Science and Engineering**, v. 18, n. 5, p. 825–836, 2002.

BERMAN, J. J. Principles of Big Data: Preparing, Sharing, and Analyzing Complex Information. In: **Principles of Big Data**. 1. ed. Burlington, MA, USA: Morgan Kaufmann, 2013. cap. Introducti, p. xix–xxvi.

BERTSEKAS, D.; GALLAGER, R. **Data networks**. 2. ed. Englewood Cliffs, NJ, USA: Prentice-Hall, Inc., 1992.

BRAAM, P. **The Lustre Storage Architecture**. 2004. Disponível em: <ftp://ftp.uni-duisburg.de/pub/linux/filesys/Lustre/lustre.pdf>.

BRAAM, P. J.; SCHWAN, P. Lustre: The intergalactic file system. In: **Ottawa Linux Symposium**. [S.l.: s.n.], 2002. p. 50.

CALZAROSSA, M.; SERAZZI, G. Workload characterization: a survey. **Proceedings of the IEEE**, v. 81, n. 8, p. 1136–1150, 1993.

CAPPELLO, F. et al. Grid'5000: a large scale and highly reconfigurable grid experimental testbed. In: **IEEE/ACM International Workshop on Grid Computing**. Seattle, WA, USA: IEEE, 2005. p. 99–106.

CARNS, P. et al. Small-file access in parallel file systems. In: **IEEE International Symposium on Parallel & Distributed Processing**. [S.l.]: IEEE, 2009. p. 1–11.

CARNS, P. et al. PVFS: A Parallel File System for Linux Clusters. In: **Linux Showcase and Conference**. [S.l.: s.n.], 2000.

CERN. **About CERN - Computing**. ago. 2014. Disponível em: <http://home.web.cern.ch/about/computing>.

COULOURIS, G. et al. **Distributed Systes: Concepts and Design**. 5. ed. Boston: Addison-Wesley, 2012. 1067 p.

CSIRO. **ASKAP Technologies: Computing**. ago. 2014. Disponível em: <http://www.atnf.csiro.au/projects/askap/computing.html>.

ELNAFFAR, S.; MARTIN, P. Techniques and a Framework for Characterizing Computer Systems' Workloads. In: **Innovations in Information Technology**. [S.l.]: IEEE, 2006. p. 1–5.

ELSEVIER. **ScienceDirect**. 2014. Disponível em: <http://www.sciencedirect.com>.

ELSEVIER. **Scopus**. 2014. Disponível em: <http://www.scopus.com>.

ESNET. **IPERF3 Benchmark**. Jul 2014. Disponível em: <http://software.es.net/iperf/>.

EVERITT, B.; HOTHORN, T. **An Introduction to Applied Multivariate Analysis with R**. 1. ed. New York: Springer New York, 2011. 274 p. (Use R!).

GRID'5000. **Grid'5000**. 2014. Disponível em: <http://www.grid5000.fr>.

GUPTA, R.; GUPTA, S.; SINGHAL, A. Big Data: Overview. **International Journal of Computer Trends and Technology (IJCTT)**, v. 9, n. 5, p. 266–268, 2014.

HALLIBURTON. **Landmark Diskos Operations**. 2014. Disponível em: <http://www.diskos.com/>.

HILDEBRAND, D.; WARD, L.; HONEYMAN, P. Large files, small writes, and pNFS. In: **Proceedings of the 20th annual international conference on Supercomputing - ICS '06**. New York, New York, USA: ACM Press, 2006. p. 116–124.

IDC. **The Digital Universe of Opportunities: Rich Data and Increasing Value of the Internet of Things**. abr. 2014. Disponível em: <http://www.emc.com/leadership/digital-universe/2014iview/executive-summary.htm>.

IEEE. **IEEE Standard for Information Technology-Portable Operating System Interface (POSIX)**. [S.l.]: IEEE, 2004. 1–91 p.

IEEE. **IEEEXplore**. 2014. Disponível em: <http://ieeexplore.ieee.org>.

INACIO, E. C.; DANTAS, M. A. R. A Survey into Performance and Energy Efficiency in HPC , Cloud and Big Data Environments. **International Journal of Networking and Virtual Organisations**, p. 1–21, 2014.

INACIO, E. C. et al. Performance Impact of Operating Systems' Caching Parameters on Parallel File Systems. In: **ACM SIGAPP Symposium on Applied Computing (SAC)**. Salamanca: ACM Press, 2015.

JANUKOWICZ, J.; EASTWOOD, M. Storage Acceleration Solving IO Performance Gap Problem. **IDC Analyst Connection**, IDC, v. 1359, p. 1–4, set. 2012.

KIMPE, D.; ROSS, R. B. **Storage Models: Past, Present, and Future**. [S.l.], 2014. 335–345 p.

KITCHENHAM, B. **Procedures for Performing Systematic Reviews**. Keele, UK, 2004. 1–28 p.

KLEIMAN, S. R. Vnodes: An Architecture for Multiple File System Types in Sun UNIX. In: **Summer USENIX Conference**. Atlanta: [s.n.], 1986. p. 238–247.

KUNKEL, J. M.; LUDWIG, T. Performance Evaluation of the PVFS2 Architecture. In: **EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing (PDP)**. Napoli, ITA: IEEE, 2007. p. 509–516.

LANEY, D. **3D Data Management: Controlling Data Volume, Velocity, and Variety**. META Group Research Note, fev. 2001. 4–6 p.

LATHAM, R.; ROSS, R. B. Parallel I/O Basics. In: **Earth System Modelling - Volume 4**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, (SpringerBriefs in Earth System Sciences, v. 4). cap. 2, p. 3–12.

LAWRENCE LIVERMORE NATIONAL LABORATORY. **IOR HPC Benchmark**. Jul 2014. Disponível em: <http://sourceforge.net/projects/ior-sio>.

LEVIN, D. V.; MCGRATH, R.; AKKERMAN, W. **strace**. Jul 2014. Disponível em: <http://sourceforge.net/projects/strace/>.

LI, X. et al. Small Files Problem in Parallel File System. In: **International Conference on Network Computing and Information Security (NCIS)**. [S.l.]: IEEE, 2011. v. 2, p. 227–232.

LIGON III, W. B.; ROSS, R. B. Implementation and performance of a parallel file system for high performance distributed applications. In: **IEEE International Symposium on High Performance Distributed Computing (HPDC)**. [S.l.]: IEEE, 1996. p. 471–480.

LIU, N. et al. Modeling a Leadership-Scale Storage System. In: WYRZYKOWSKI, R. et al. (Ed.). **Parallel Processing and Applied Mathematics**. [S.l.]: Springer Berlin Heidelberg, 2012. p. 10–19.

LIU, Y.; FIGUEIREDO, R. Towards Simulation of Parallel File System Scheduling Algorithms with PFSsim. In: **IEEE International Workshop on Storage Network Architecture and Parallel I/O (SNAPI)**. [S.l.: s.n.], 2011.

LU, Y. et al. A New Data Sieving Approach for High Performance I/O. In: HIUK, J. et al. (Ed.). **Future Information Technology (FutureTech)**. [S.l.]: Springer Netherlands, 2012. (Lecture Notes in Electrical Engineering, v. 164), p. 111–121.

MACEDO, D. D. J. de. **Um Modelo Distribuído de Armazenamento Hierárquico de Conhecimento Médico**. Tese (Doutorado) — Universidade Federal de Santa Catarina, 2014.

MACEDO, D. D. J. de et al. An architecture for DICOM medical images storage and retrieval adopting distributed file systems. **International Journal of High Performance Systems Architecture**, Inderscience Publishers, v. 2, n. 2, p. 99, mar. 2009.

MATHUR, A. et al. The new ext4 filesystem: current status and future plans. In: **Ottawa Linux Symposium**. Ottawa, ON: [s.n.], 2007. p. 21–33.

MICROSOFT. **Azure: Microsoft's Cloud Platform**. Jul 2014. Disponível em: <http://azure.microsoft.com/en-us/>.

MOLINA-ESTOLANO, E. et al. Building a parallel file system simulator. **Journal of Physics: Conference Series**, v. 180, n. 1, jul. 2009.

MOORE, M. et al. OrangeFS: Advancing PVFS. In: **File and Storage Technologies, USENIX Conference on. (FAST)**. San Jose, CA: USENIX Association, 2011. p. 1–2.

MPI FORUM. **MPI-2: Extensions to the Message-Passing Interface**. 1997. Disponível em: <http://mpi-forum.org/docs/mpi-2.0/mpi-20-html/mpi2-report.htm>.

MUTHITACHAROEN, A. et al. Ivy: a read/write peer-to-peer file system. **ACM SIGOPS Operating Systems Review**, ACM, v. 36, p. 31–41, dez. 2002.

NGUYEN, H. Q.; APON, A. Hierarchical performance measurement and modeling of the linux file system. In: **WOSP/SIPEW International Conference on Performance Engineering (ICPE)**. New York, New York, USA: ACM Press, 2011. v. 36, n. 5, p. 73–84.

NGUYEN, H. Q.; APON, A. Parallel file system measurement and modeling using colored petri nets. In: **International Conference on Performance Engineering (ICPE)**. New York, New York, USA: ACM Press, 2012. p. 229–240.

OBAIDAT, M. S.; BOUDRIGA, N. A. **Fundamentals of Performance Evaluation of Computer and Telecommunications Systems**. 1. ed. Hoboken, NJ, USA: John Wiley & Sons, 2010.

ORANGEFS. **Orange File System**. Jul 2014. Disponível em: <http://orangefs.org>.

PETERSON, L. L.; DAVIE, B. S. **Computer Networks: A Systems Approach, 3rd Edition**. 3. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003.

PINKSTON, T. M.; DUATO, J. Computer architecture: a quantitative approach. In: _____. 4. ed. [S.l.]: Morgan Kaufmann, 2006. cap. Interconnection Networks.

SAINI, S. et al. I/O performance characterization of Lustre and NASA applications on Pleiades. In: **International Conference on High Performance Computing (HiPC)**. Pune, India: IEEE, 2012. p. 1–10.

SANDBERG, R. et al. Design and implementation of the sun network filesystem. In: **Summer USENIX Conference**. [S.l.: s.n.], 1985. p. 119–130.

SATO, K. et al. Design and modeling of a non-blocking checkpointing system. In: **International Conference on High Performance Computing, Networking, Storage and Analysis (SC)**. [S.l.]: IEEE Computer Society Press, 2012. p. 1–10.

SCHMUCK, F.; HASKIN, R. GPFS: A Shared-Disk File System for Large Computing Clusters. In: **File and Storage Technologies, USENIX Conference on. (FAST)**. [S.l.: s.n.], 2002. p. 1–14.

SETTLEMYER, B. W. **A Study of Client-side Caching in Parallel File Systems**. Tese (Doutorado) — Clemson University, Clemson, South California, USA, 2009.

SHAN, H.; ANTYPAS, K.; SHALF, J. Characterizing and predicting the I/O performance of HPC applications using a parameterized synthetic benchmark. In: **ACM/IEEE Conference on Supercomputing (SC)**. Piscataway, NJ, USA: IEEE Press, 2008.

SILBERSCHATZ, A.; GALVIN, P. B.; GAGNE, G. **Operating System Concepts**. 8. ed. Hoboken, NJ, USA: Wiley Publishing, 2008.

SOARES, T. S. **Uma Arquitetura Paralela Para o Armazenamento de Imagens Médicas em Sistemas de Arquivos Distribuídos**. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2012.

SOARES, T. S. et al. An Approach Using Parallel Architecture to Storage DICOM Images in Distributed File System. **Journal of Physics: Conference Series**, IOP Publishing, v. 341, n. 1, p. 1–8, fev. 2012.

SOARES, T. S. et al. PH5WRAP : A Parallel Approach To Storage Server of Dicom Images. In: **International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)**. [S.l.: s.n.], 2012.

SONG, H. et al. A cost-intelligent application-specific data layout scheme for parallel file systems. In: **International Symposium on High Performance Distributed Computing (HPDC)**. New York, NY, USA: ACM Press, 2011. p. 37–48.

SONG, H. et al. Cost-intelligent application-specific data layout optimization for parallel file systems. **Cluster Computing**, v. 16, n. 2, p. 285–298, fev. 2012.

SUN, X.-H.; CHEN, Y.; YIN, Y. Data layout optimization for petascale file systems. In: **4th Petascale Data Storage Workshop, Proceedings of the (PDSW)**. New York, New York, USA: ACM Press, 2009. p. 11–15.

TANENBAUM, A. S. **Modern Operating Systems**. 3. ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2007.

TANENBAUM, A. S.; STEEN, M. V. **Distributed Systems: Principles and Paradigms**. 2. ed. Amsterdam: Pearson Prentice Hall, 2006.

THAKUR, R. et al. Passion: Optimized I/O for Parallel Applications. **Computer**, IEEE Computer Society Press, v. 29, n. 6, p. 70–78, jun. 1996.

THE HDF GROUP. **Hierarchical Data Format, version 5**. 1997. Disponível em: <http://www.hdfgroup.org/HDF5/>.

THOMSON REUTERS. **Web of Science**. 2014. Disponível em: <http://apps.webofknowledge.com>.

TOP500. **TOP500 The List**. Nov 2014. Disponível em: <http://www.top500.org/>.

VELHO, P. et al. On the validity of flow-level tcp network models for grid and cloud simulations. **ACM Transactions on Modeling and Computer Simulation**, ACM, v. 23, n. 4, p. 1–26, out. 2013.

WEIL, S. A. et al. Ceph: a scalable, high-performance distributed file system. In: **Operating Systems Design and Implementation (OSDI)**. Berkeley, CA, USA: USENIX Association, 2006. p. 307–320.

WIEERS, D. **Dstat: Versatile resource statistics tool**. Jul 2014. Disponível em: <http://dag.wiee.rs/home-made/dstat/>.

YU, W.; VETTER, J. S.; ORAL, H. S. Performance characterization and optimization of parallel I/O on the cray XT. In: **IEEE International Parallel and Distributed Processing Symposium**. Miami, Floripa, USA: IEEE, 2008.

ZHAO, T.; HU, J. Performance Evaluation of Parallel File System Based on Lustre and Grey Theory. In: **2010 Ninth International Conference on Grid and Cloud Computing**. [S.l.]: IEEE, 2010. p. 118–123.

ZHAO, T. et al. Evaluation of a Performance Model of Lustre File System. In: **2010 Fifth Annual ChinaGrid Conference**. [S.l.]: IEEE, 2010. p. 191–196.

APÊNDICE A – Publicações

A.1 PUBLICAÇÕES

Durante a realização dessa pesquisa, alguns trabalhos foram produzidos. Uma revisão bibliográfica sobre a utilização do processo de caracterização de carga de trabalho para melhoria de desempenho e de eficiência energética em sistemas de computação de alto desempenho, nuvens computacionais e *Big Data*, resultou na seguinte publicação:

Título: A Survey into Performance and Energy Efficiency in HPC, Cloud and Big Data Environments.
Autores: Eduardo Camilo Inacio e Mario Antonio Ribeiro Dantas.
Periódico: International Journal of Networking and Virtual Organisations (IJNVO).
Ano: 2014.
Extrato QUALIS: B3.

Um resumo estendido apresentando a caracterização e modelagem do efeito da cache dos sistemas operacionais utilizados nos nodos de armazenamento sobre o desempenho geral do sistema de arquivos paralelo resultou na seguinte publicação:

Título: Performance Impact of Operating Systems' Caching Parameters on Parallel File Systems.
Autores: Eduardo Camilo Inacio, Francieli Zanon Boito, Douglas Dyllon Jerônimo de Macedo, Mario Antonio Ribeiro Dantas e Philippe Olivier Alexandre Navaux.
Conferência: ACM/SIGAPP Symposium on Applied Computing (Special Track on Operating Systems).
Ano: 2015.
Extrato QUALIS: A1.

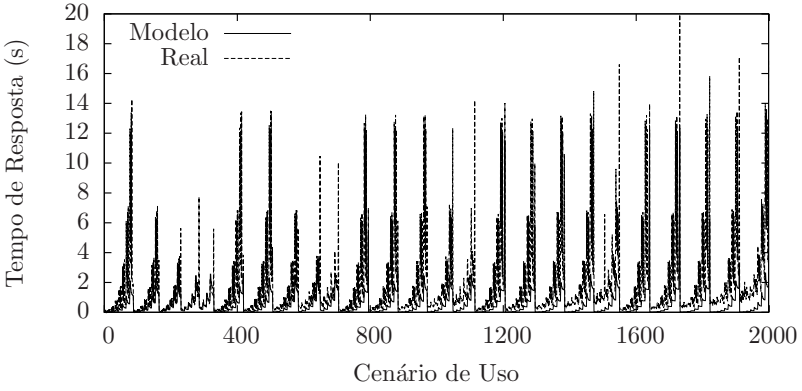
APÊNDICE B – Resultados Adicionais

B.1 RESULTADOS ADICIONAIS

Na Seção 5.3 são apresentados os resultados da avaliação do modelo proposto. Em virtude do grande número de resultados, a apresentação destes foi dividida com relação aos três casos considerados para o cálculo do tempo de resposta (ver Seção 4.2). Para a argumentação de cada caso, foram utilizados os resultados de um dos ambientes experimentais. Neste apêndice, são apresentados resultados da avaliação complementares aos da Seção 5.3.

Para a avaliação do caso 1, em que as caches dos nodos de armazenamento suportam o volume de dados enviados para a escrita, foram utilizados os resultados pertinentes ao cluster Grid'5000 Graphene. Os resultados para o mesmo caso nos ambientes criados na Microsoft Azure e no LAPESD são apresentados, respectivamente, nas Figuras 24 e 25.

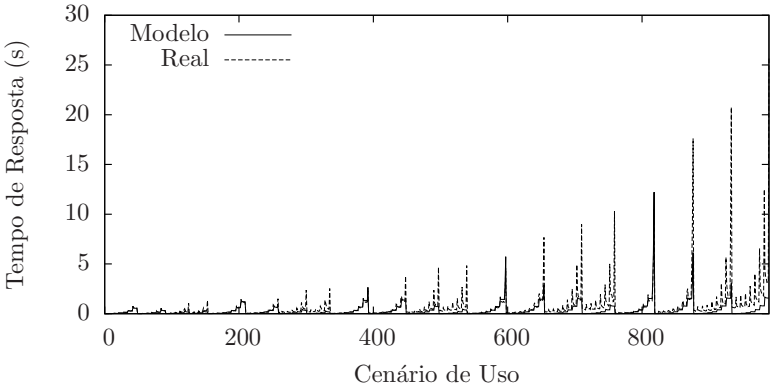
Figura 24: Comparação do desempenho estimado pelo modelo e medido no ambiente Microsoft Azure para cenários no caso 1.



Observa-se nos resultados dos dois ambientes uma similaridade entre o comportamento representado pelo modelo e o observado experimentalmente. Em termos de valores absolutos, os resultados indicam que o modelo subestima o tempo de resposta fornecido pelo ambiente. Conforme discutido anteriormente, uma das causas para o tempo de resposta medido nos ambientes Microsoft Azure e LAPESD ser superior ao tempo estimado pelo modelo é o compartilhamento de recursos físicos entre os nodos de armazenamento. Esse efeito é melhor obser-

vado nos resultados do ambiente LAPESD (Figura 25) para os cenários de uso (eixo x) mais a direita, que são os cenários com maior número de nodos de armazenamento.

Figura 25: Comparação do desempenho estimado pelo modelo e medido no ambiente LAPESD para cenários no caso 1.



Os resultados do ambiente LAPESD foram utilizados nesta dissertação para apresentar a avaliação do modelo para o caso 2: caso em que o volume de dados enviado para o sistema de arquivos paralelo supera o espaço da cache de escrita. Nas Figuras 26 e 27 são apresentados os resultados referentes ao mesmo caso no cluster Grid'5000 Graphene e no ambiente Microsoft Azure.

Nos resultados do caso 2 para o cluster Grid'5000 Graphene (Figura 26) são observados significativos desvios em termos de valores absolutos estimados pelo sistema, confirmando o erro percentual elavado apresentado na Tabela 4. Em termos de representação do comportamento das curvas de desempenho, observa-se que pontos discretos dos resultados experimentais são aparentemente simplificados pelo modelo. A explicação para estes resultados requer uma análise pontual neste ambiente e deverá ser conduzida em trabalhos futuros.

A representação do caso 2 no ambiente Microsoft Azure (Figura 27), por outro lado, foi a que obteve o menor erro percentual médio neste ambiente. Percebe-se nos resultados a similaridade na representação do comportamento e a aproximação dos valores absolutos estimados pelo modelo.

Por fim, a avaliação do terceiro caso foi apresentada nesta dissertação utilizando os resultados do ambiente Microsoft Azure. O terceiro

Figura 26: Comparação do desempenho estimado pelo modelo e medido no cluster Grid'5000 Graphene para cenários no caso 2.

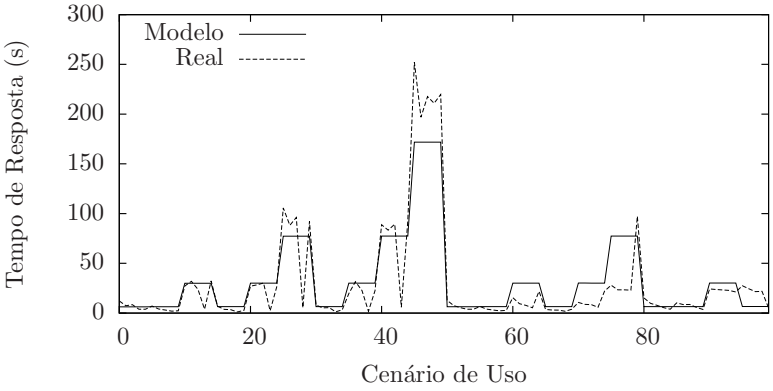
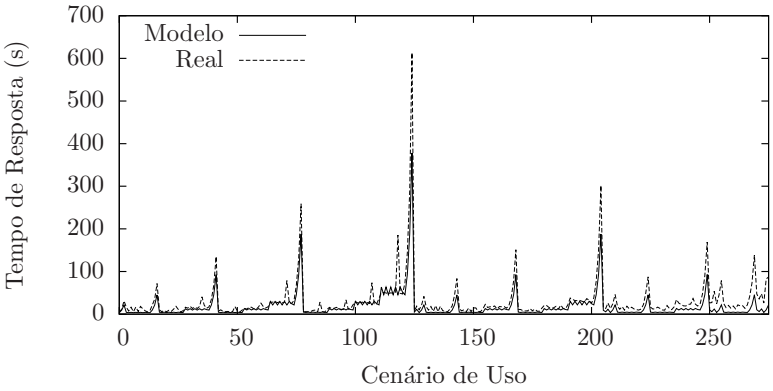


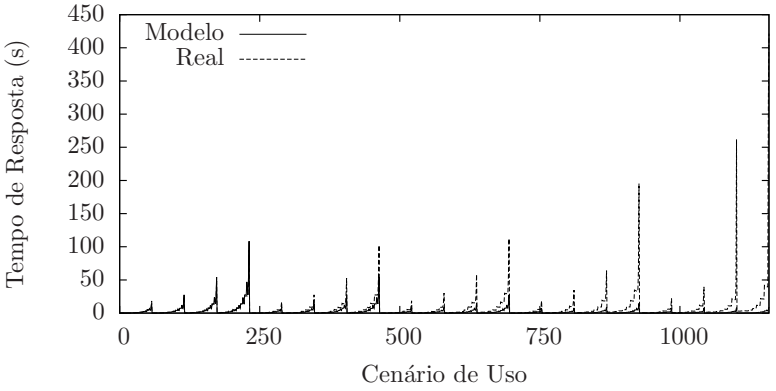
Figura 27: Comparação do desempenho estimado pelo modelo e medido no ambiente Microsoft Azure para cenários no caso 2.



caso refere-se aos cenários em que o cliente invoca explicitamente a sincronização de dados nos dispositivos de armazenamento por meio da chamada de sistema `fsync`. A Figura 28 apresenta a avaliação deste caso para o ambiente LAPESD.

Assim como nos casos anteriores, percebe-se que o modelo é capaz de representar o comportamento do desempenho do sistema para uma significativa quantidade de cenários de uso. Em termos da esti-

Figura 28: Comparação do desempenho estimado pelo modelo e medido no ambiente LAPESD para cenários no caso 3.



mativa de valores absolutos, percebe-se que a diferença entre o valor predito e medido aumenta de esquerda para a direita no eixo x. Esse efeito foi descrito na argumentação do caso 2 da Seção 5.3, utilizando os resultados deste mesmo ambiente como exemplo. Trata-se da concorrência de acesso aos dispositivos de armazenamento, provocado pelo processo de sincronização. Como a quantidade de nodos de armazenamento aumenta da esquerda para a direita no eixo x, os cenários de uso mais a direita apresentam um maior nível de concorrência e, por consequência, um maior atraso na sincronização.