

## METODOLOGIA DE AVALIAÇÃO OPERACIONAL PARA SISTEMAS DE COMPUTAÇÃO CIENTÍFICA DISTRIBUÍDA DE ALTO DESEMPENHO

**Mariza Ferro**

**Bruno Schulze**

**Antônio R. Mury**

Laboratório Nacional de Computação Científica - LNCC  
Av. Getúlio Vargas, 333 - Quitandinha, Petrópolis - RJ  
mariza,schulze, aroberto@lncc.br

### RESUMO

A Computação Massivamente Paralela e Distribuída é fundamental para alavancar o progresso científico em inúmeras áreas da ciência. Porém, cada domínio de investigação tem aplicações com diferentes requisitos computacionais, os quais dependem da definição adequada desses sistemas de alto desempenho para se obter a eficácia e eficiência na resolução dos seus problemas. Assim, o pesquisador se depara com decisões complexas sobre a escolha da melhor infraestrutura para a execução do seu conjunto de aplicações científicas. Além disso, os métodos tradicionais de avaliação de desempenho por meio da execução de benchmarks possuem inúmeras limitações. Com o objetivo de superar essas dificuldades e limitações é que este trabalho apresenta uma metodologia de avaliação operacional que orienta pesquisadores e técnicos na aquisição e manutenção de equipamentos de computação de alto desempenho, voltados aos requisitos das suas aplicações. Foram investigados os comportamentos e requisitos computacionais exigidos para diferentes aplicações e sua combinação com arquiteturas, modelos de programação, tamanhos de problema e as relações com o desempenho. O conhecimento obtido viabilizou o desenvolvimento da metodologia, que foi avaliada por meio de um estudo de caso, no qual suas contribuições foram mensuradas.

**Palavra-chave:** Avaliação Operacional; Computação de Alto Desempenho; Computação Científica; Tomada de Decisão.

### ABSTRACT

Scientific Computing typically requires large computational needs which have been addressed with High Performance Distributed Computing. It is essential to efficiently deploy a number of complex scientific applications, which have different characteristics, and so require distinct computational resources too. However, in many research laboratories, this high performance architecture is not dedicated. So, the architecture must be shared to execute a set of scientific applications, with so many different execution times

and relative importance to research. Also, the high performance architectures have different characteristics and costs. When a new infrastructure has to be acquired to meet the needs of this scenario, the decision-making is hard and complex. In this work, we present a Gain Function as a model of an utility function, with which it is possible a decision-making with confidence. With the function is possible to evaluate the best architectural option taking into account aspects of applications and architectures, including the executions time, cost of architecture, the relative importance of each application and also the relative importance of performance and cost on the final evaluation. This paper presents the Gain Function, examples, and a real case showing their applicabilities.

**Keywords:** Operational Analysis; High Performance Computing; Scientific Computing; Decision Making.

**Como Citar:**

FERRO, Mariza; MURY, Antônio R., SCHULZE, Bruno. METODOLOGIA DE AVALIAÇÃO OPERACIONAL PARA SISTEMAS DE COMPUTAÇÃO CIENTÍFICA DISTRIBUÍDA DE ALTO DESEMPENHO. In: SIMPÓSIO DE PESQUISA OPERACIONAL E LOGÍSTICA DA MARINHA, 19., 2019, Rio de Janeiro, RJ. *Anais* [...]. Rio de Janeiro: Centro de Análises de Sistemas Navais, 2019.

## 1. INTRODUÇÃO

A avaliação e os testes para determinar a utilidade de um novo invento é um conceito bastante antigo. Porém, conforme os equipamentos e sistemas se tornaram cada vez mais complexos, também surgiu a necessidade de evolução das metodologias de avaliação e teste. Assim, as avaliações passaram a ser realizadas iniciando-se por uma especificação detalhada do sistema com a descrição detalhada de cada característica e o estabelecimento de critérios que satisfizessem às exigências das características descritas. O passo final consistia em testar formalmente o sistema, para determinar se cada critério tinha sido satisfeito. A avaliação dos resultados de teste versus critérios estabelecidos determinava se o sistema era satisfatório. Porém, logo ficou evidente que essa metodologia era insatisfatória. Quase sempre os sistemas satisfaziam as especificações, isto é, passavam nos testes mas não desempenhavam satisfatoriamente as funções pretendidas quando sua utilização saía dos laboratórios de testes e passava a ser utilizada no cotidiano das pessoas. Por exemplo, um automóvel pode satisfazer a cada uma de centenas de especificações e, mesmo assim, perder seu mercado, se os consumidores considerarem-no desconfortável e difícil de dirigir [1]. Qualquer procedimento de avaliação está sujeito a falhas, entretanto, elas não eram tão evidentes até o desenvolvimento do enfoque sistêmico, que considera o sistema como um todo, sob o ponto de vista do objetivo a ser cumprido. Quando essa técnica foi aplicada na avaliação, ficou claro que era necessário desenvolver-se uma nova técnica, em que o sistema, incluindo homens e equipamentos, fosse avaliado quando em operação em ambiente tão real quanto possível, procurando-se determinar o quão bem ele pode cumprir o seu objetivo. Chama-se essa técnica de Avaliação Operacional (AO).

O mesmo conceito se aplica na avaliação de Sistemas de Computação de Alto Desempenho (HPC). A maneira tradicional de se fazer a avaliação deste tipo de sistema ainda é por meio de testes com *benchmarks*, que quando utilizados, muitas vezes estão dissociados das características das aplicações e das cargas de trabalho usualmente utilizadas. Segundo [2] é muito comum que usuários de sistemas de HPDC se decepcionem com o de-

sempenho obtido com seus equipamentos após a aquisição por terem utilizado *benchmarks* não representativos das suas necessidades. Devido à grande complexidade de parâmetros envolvidos na avaliação de HPC, somados as características das próprias aplicações, é fundamental que haja uma metodologia que guie o tomador de decisão (o pesquisador) a conhecer as características das suas aplicações e a definir quais são os parâmetros relevantes que lhe permitam chegar a uma avaliação de desempenho mais confiável. Além disso, em casos bastante frequentes, não é possível obter um equipamento totalmente dedicado a execução de uma única aplicação. Nesses casos é possível ter uma combinação de classes de aplicações diferentes, com requisitos computacionais distintos, e é preciso mensurar qual o equipamento que oferece melhor ganho para um conjunto de aplicações, tornando a avaliação e decisão mais complexas.

Assim, na computação também se torna evidente que é preciso evoluir os procedimentos de avaliação e testes, incluindo aplicações e equipamentos, e avaliando assim o sistema de HPC em operação em ambiente tão real quanto possível, procurando determinar o quão bem ele pode cumprir o seu objetivo (executar o conjunto de aplicações científicas). Surge assim a necessidade de uma Metodologia de Avaliação Operacional para Sistemas de Computação Científica Distribuída de Alto Desempenho. Essa metodologia, cuja sistemática é desenvolvida e apresentada resumidamente neste trabalho, direciona o pesquisador nessa avaliação, permitindo mensurar qual a melhor arquitetura, para uma ou várias aplicações, considerando aspectos específicos das aplicações, das arquiteturas e seus custos.

A metodologia foi avaliada em um estudo de caso na bioinformática cujo foco foi a execução de três aplicações de bioinformática, em tempo aceitável de execução. Foram avaliadas duas arquiteturas diferentes de alto desempenho e os resultados apontaram o ganho de cada arquitetura para o problema em questão. O estudo de caso foi focado na avaliação, principalmente da fase final da AO, e possibilitou a validação e a verificação da função de ganho e da utilidade da metodologia e das suas contribuições para os pesquisadores de outras áreas.

## 2. FUNDAMENTAÇÃO TEÓRICA

Exercendo hoje um papel fundamental no processo científico, a Computação Massivamente Paralela e Distribuída (HPDC)<sup>1</sup> é um segmento da ciência da computação que tem como objetivo a melhoria do desempenho de aplicações distribuídas e paralelas, utilizando complexas infraestruturas computacionais [3]. Com ela é possível o processamento de grandes volumes de dados experimentais criados por novas gerações de instrumentos e aplicações científicas, as quais exigem grande largura de banda, redes de baixa latência e recursos computacionais de alto desempenho [4]. Esse tipo de infraestrutura tem se tornado crucial para as pesquisas científicas em muitos domínios de investigação, tais como a bioinformática, a dinâmica de fluidos, a medicina assistida por computação, entre outras.

Entretanto, cada domínio de investigação tem aplicações com diferentes requisitos computacionais, os quais dependem da definição adequada dos sistemas de HPDC envolvidos para se obter a eficácia e eficiência na resolução dos seus problemas. Assim, o tomador de decisão (o pesquisador) se depara com decisões complexas e que na maioria das vezes não fazem parte da sua especialidade, pois avaliar o desempenho das aplicações

<sup>1</sup>Do inglês High Performance and Distributed Computing.

em tais infraestruturas não é tarefa trivial.

O atual paradigma utilizado para se determinar o desempenho de uma infraestrutura computacional é por meio de programas de avaliação de desempenho, os *benchmarks*. Porém, com o seu uso normalmente o que se obtém são picos teóricos de desempenho, uma medida pouco informativa e que pouco auxilia quando é necessário determinar o real desempenho da infraestrutura frente aos requisitos das aplicações. O que se percebe é que pela falta de uma metodologia que possibilite essa avaliação de maneira formal, a opção feita hoje pela maioria dos pesquisadores é adquirir uma arquitetura baseada no máximo em Flops (obtidas justamente pela execução dos *benchmarks*) ou aquela que ofereça o maior número de nós de processamento. Porém, quando essas infraestruturas são utilizadas em seu ambiente de pesquisa, raramente os picos teóricos apresentados pelos fabricantes são atingidos, e muitas vezes os gastos financeiros não se concretizam em bons desempenhos [5].

É evidente que é necessário mudar esse paradigma, criando procedimentos de avaliação que considerem as características das aplicações, cargas de trabalho e equipamentos, avaliando assim o sistema de HPDC em operação em ambiente tão real quanto possível. Precisa-se criar formas sistemáticas de guiar o pesquisador pela definição correta, desde os objetivos da avaliação, aos parâmetros relevantes de serem considerados para o seu problema, até a realização de testes, voltados para suas aplicações.

Para alcançar o objetivo proposto, o ponto de partida é entender o comportamento das aplicações científicas, as quais têm diferentes requisitos computacionais, pois seus algoritmos e métodos matemáticos utilizados na computação científica variam de um domínio para outro. Por isso, entender suas características e criar padrões de comportamento que sejam capazes de representar uma ampla gama de aplicações se tornou uma busca tanto para projetistas de novas arquiteturas paralelas, como os de softwares e aqueles interessados nas suas otimizações. Um esforço relevante nessa busca foi feito por [6], criador de uma categorização que consegue representar o comportamento das aplicações em termos de processamento e movimento de dados para um grande número de domínios. Essa categorização, conhecida como Dwarfs (ou *motifs*), contribuiu para o estudo das aplicações científicas neste trabalho, pois se mostra uma forma confiável de classificação, e de conduzir os experimentos de forma coerente, já que continua em utilização e desenvolvimento por grandes projetos em importantes laboratórios de pesquisa [7] [8] Coral:2013 [9].

Baseando-se nessa classificação, foi investigado quais os principais aspectos que permitem categorizar as aplicações científicas em termos de seus requisitos computacionais e como estes se relacionam com diferentes arquiteturas de alto desempenho e parâmetros, tais como modelos de programação e cargas de trabalho.

Outro referencial teórico importante utilizado como base ao desenvolvimento deste trabalho é a Análise Operacional (AO). O objetivo da AO é justamente ser um método sistemático de análise que auxilia no processo de tomada de decisões quanto à obtenção, ao emprego e às modificações do sistema avaliado. A AO procura estimar a eficácia e a adequabilidade operacional do sistema por meio de experimentos controlados, onde se busca o maior realismo possível. As motivações que levaram a escolha da AO como metodologia de referência é por sua consagração em avaliar sistemas complexos no âmbito das tomadas de decisões militares, industriais e governamentais. Além disso, a Teoria da Utilidade Multiatributo (MAUT) foi empregada para o desenvolvimento matemático da metodologia, como importante auxiliar para o problema de decisão multicritério. As-

sim, esse referencial permitiu desenvolver a metodologia apresentada neste trabalho, para sistemas computacionais, seguindo sua rígida sistemática de aplicação e avaliação de resultados.

## 2.1. TRABALHOS RELACIONADOS

Há tempos, trabalhos desenvolvidos para modelagem do desempenho de sistemas computacionais têm sido explorados como forma de facilitar as decisões de novos projetos e planejar a capacidade dos sistemas. Atualmente, com o desenvolvimento acelerado de microprocessadores, diversos estudos se dedicam a explicar e criar novos modelos para avaliação de desempenho, terminologias e técnicas para projetistas e usuários. Exemplos são os trabalhos de [5] e [10] os quais apresentam as mais diversas técnicas e métricas para modelagem e avaliações de desempenho que variam desde abordagens estatísticas, de simulação, de monitoramento e de planejamento até o estudo de diversos tipos de *benchmarks*. Porém, dada a abrangência do assunto, a apresentação de trabalhos relacionados é limitada aos trabalhos que utilizem AO para avaliação de sistemas computacionais.

A abordagem da AO como metodologia específica para computação tem sido usada na modelagem e predição de desempenho de sistemas computacionais, principalmente para Teoria de Filas e para a análise de restrições, identificando os dispositivos que possam limitar o desempenho do sistema como um todo. O emprego da AO aplicada ao problema de entender e de prever o desempenho de aplicações e sistemas de computação foi proposta inicialmente por [11]. A partir de então, foram caracterizados os problemas do mundo real que poderiam ser tratados com técnicas operacionais e derivadas leis e teoremas operacionais dando respostas exatas para uma grande classe de problemas práticos de desempenho.

Resultados elementares incluem as clássicas Leis Operacionais (por exemplo *Little's Law*) consideradas um poderoso e versátil resultado de modelagem de desempenho e que serviu de base para outros métodos mais sofisticados apresentados posteriormente. [12] desenvolveram diversos trabalhos para a criação de modelos para teoria de filas [13], [14].

O trabalho de [15] aplica AO ao problema de prever e entender o desempenho das aplicações em servidores paralelos. Foram desenvolvidas novas leis operacionais que inferem o atraso de filas e limitam as alterações de desempenho quando a capacidade do servidor é alterada. Com isso, permite o planejamento da capacidade e o provisionamento dinâmico de recursos.

No estudo de [16] é utilizada AO para realizar a análise de desempenho das aplicações diante de mudanças na velocidade do processador. Foram derivadas leis operacionais que permitem limitar os atrasos das filas (em nível de aplicação) que podem acontecer se uma mesma carga de trabalho fosse executada por um processador mais rápido ou mais lento. Tais limites auxiliam nas tomadas de decisão de escolha de hardware, planejamento de capacidade e até do controle de gasto de energia.

[17] desenvolveu um modelo para classificar e controlar o risco das previsões derivadas de modelos probabilísticos de cargas de processamento utilizando entre outras teorias, a AO. A proposta emprega uma caracterização alternativa de incerteza, a qual é expressa exclusivamente em termos de propriedades observáveis. Segundo o autor isto proporciona uma série de vantagens sobre os modelos probabilísticos tradicionais, além de

acarretar uma ligação mais direta com os problemas que surgem em situações práticas, oferecendo uma nova perspectiva sobre a análise de risco e um novo método para melhorar a eficiência de simulações por computador.

Foi realizada uma extensa pesquisa bibliográfica, porém não foram encontrados outros trabalhos que utilizassem a AO para avaliação de sistemas computacionais como um todo. Todos os projetos apresentados analisam componentes individuais dos sistemas computacionais e não levam em consideração aspectos das aplicações que possam ser utilizadas nesses ambientes e nem como estes componentes ao interagir se comportam e contribuem ou não para este sistema complexo. Entretanto, nesta tese o sistema de HPDC é analisado levando-se em consideração todos os componentes e as aplicações associadas.

### 3. METODOLOGIA PROPOSTA

A metodologia desenvolvida compreende uma sequência de fases e passos que orienta toda realização da avaliação. Na Figura 1 é possível ver um detalhamento de cada uma das fases, seus respectivos passos e resultados gerados no formato de documentos.

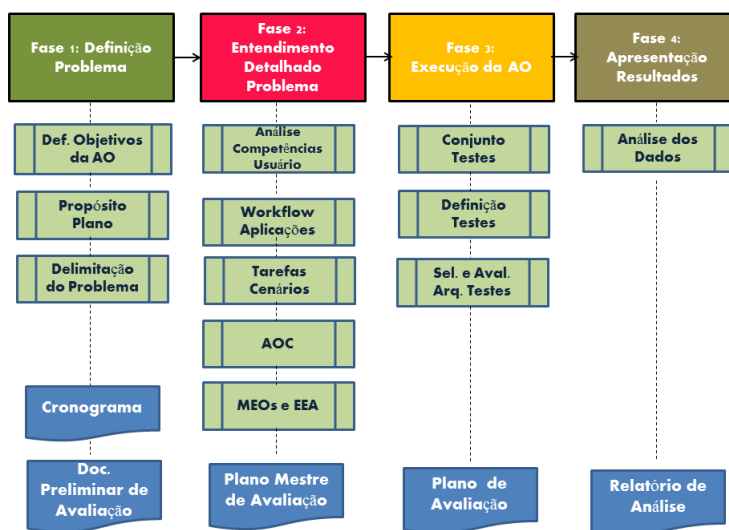


Figura 1: Metodologia de AO para HPDC - Sequência de Fases e Passos.

O início formal da aplicação da metodologia acontece após a definição, pelo grupo de pesquisa, de que o melhor para atingir seus objetivos naquele momento é a obtenção ou modernização de seus sistemas, e que para cumprir esses propósitos o mais adequado é por meio da aplicação da metodologia proposta pela AO.

Após essa definição, inicia-se então a fase de **definição do problema**. Nessa fase procura-se obter o enunciado do problema que se deseja resolver e todos os elementos relevantes para sua solução, de modo claro para todos os envolvidos. A determinação clara dos objetivos e propósitos específicos ligados ao processo da AO, será de fundamental importância para o planejamento das demais fases e a execução da AO pretendida. Apesar de parecer óbvia a necessidade da descrição clara de qual o objetivo da AO, geralmente há equívocos sobre a definição de quais são os reais objetivos. Os pré-requisitos são determinados nesta fase e permitirão a realização de uma modelagem inicial, ainda totalmente

conceitual gerando como resultado final um Cronograma e um Documento Preliminar de Avaliação.

Com o Documento Preliminar de Avaliação em mãos, segue-se para fase seguinte (**Fase 2: Entendimento Detalhado do Problema**) na qual o problema do usuário é detalhado. Os passos dessa fase passam a ser eminentemente técnicos e se destinam a preparar a execução, propriamente dita, da AO. Todo conhecimento disponível sobre o *workflow* de aplicações do usuário <sup>2</sup> e dos estudos realizados devem ser consolidados no Plano Mestre de Avaliação.

Nesse passo serão identificados as *Tarefas, Aplicações e Cenários (TAC)*, onde as tarefas irão indicar os requisitos para o sistema mais adequado, e como as técnicas de AO recomendam avaliar o sistema em condições tão reais quanto possível, são identificados os cenários, assumindo os mesmos que são previstos para o funcionamento do sistema. Para isso é preciso conhecer cada uma das tarefas que o sistema vai executar, as quais serão formalizadas em termos das aplicações a serem executadas pelo sistema.

A partir da análise das TAC serão identificados os *Aspectos Operacionais Críticos (AOC)* pertinentes ao contexto. Muitos sistemas degradaram-se quando colocados em operação, porque não foram determinados os aspectos críticos diretamente envolvidos com o cumprimento da função esperada para eles. Por exemplo, o sistema foi avaliado com um conjunto de *benchmarks* que não refletem as aplicações científicas que são predominantemente executadas pelo sistema. Quando os aspectos críticos são apropriadamente definidos, as deficiências do sistema podem ser descobertas e corrigidas e, igualmente importante, procedimentos e atitudes podem ser otimizados antes do sistema entrar em uso operacional. É importante salientar que os AOC representam os pontos em que se não forem bem tratados e houver falha, o sistema tem sua eficiência degradada. No caso em que a aplicação do usuário puder ser mapeada para alguma das classes Dwarfs de aplicação, isso já vai permitir identificar alguns AOC, que são as características próprias das classes de aplicações, as quais podem limitar o desempenho do sistema. Por exemplo, é importante não executar duas classes que tenham o mesmo AOC em paralelo, pois pode limitar demais o desempenho do sistema, muitas vezes levando a falhas.

Ainda nesta fase, o último passo consiste na decomposição dos Aspectos Operacionais Críticos visando a identificação de dados e medidas básicas relacionadas a eles, aos quais denomina-se *Elementos Essenciais de Análise* que, criteriosamente recombinaos, poderão fornecer respostas ou soluções relacionadas aos referidos aspectos, o que vem a constituir as *Medidas de Eficácia Operacional*. os EEA estabelecem um método de decomposição de aspectos críticos até o ponto onde dados e medidas de testes reais podem ser definidos. Eles compreendem um conjunto de valores operacionais básicos que devem ser medidos para realizar a análise e a avaliação necessária para fornecer respostas aos AOC. Cada valor do EEA é obtido pela coleta qualitativa ou quantitativa específica de dados testes. Um conjunto de EEA foi identificado para essa metodologia durante o desenvolvimento do trabalho, a partir do conjunto experimental. Alguns exemplos são: Tempo Total de Execução (o tempo entre o início da execução de uma aplicação e sua conclusão), Classe de aplicação (a classe Dwarf correspondente a aplicação, convertida usando valores operacionais dos demais EEA), Linguagem de Programação (a linguagem de programação utilizada para implementação da aplicação), Tempo de Uso de CPU (o tempo total que

<sup>2</sup>O termo *workflow* de aplicações é utilizado neste trabalho apenas no sentido de conhecer as aplicações e a sequência de entrada e saída de dados entre elas e não na utilização de ferramentas formais de *workflows* científicos

o processador está executando a aplicação, não incluindo o tempo que fica aguardando por tarefas de E/S ou executando outras aplicações em paralelo). A relação completa das EEAs pré-definidas estão em [18].

Tais dados básicos podem ser recombinados de alguma forma para fornecerem respostas aos aspectos críticos. Estas combinações de dados e medidas básicas constituem as MEOs. Elas são importantes, porquanto determinam como os resultados dos experimentos serão julgados, já que as MEO tem o objetivo de quantificar o grau com que dado sistema, em condições de operação tão real quanto possível, atingem sua finalidade. Além disso, com o conjunto de MEOs predefinidos é possível compreender os resultados dos testes e as conclusões acerca do sistema em vários níveis de decisão e permitir que dados de um teste sejam usados em outro teste, tanto para o mesmo sistema, quanto para outro similar. Novas MEOS e EEA podem ser definidos para uma AO específica, pois aspectos não estudados podem surgir, principalmente pela constante evolução das arquiteturas de HPDC.

Com o Plano Mestre de Avaliação pronto, segue-se para fase de execução propriamente dita da AO. Na fase de **Execução da AO**, a qual compreende a condução dos experimentos em condições de operação do sistema, tão reais quanto possível nas arquiteturas selecionadas para os testes, os quais foram definidos também nesta fase. A *definição dos conjuntos de testes* que serão executados vai depender dos EEA que já foram estabelecidos e precisam ter seus valores operacionais coletados para as análises posteriores utilizando as MEOs e determinando quais os equipamentos mais adequados para o problema. Apesar do ideal ser executar os testes em todos os tipos de arquiteturas disponíveis no mercado, visto que restrições existem, no passo de *Seleção e Avaliação dos Ambientes de Testes* são selecionados e avaliados os ambientes de teste (sistemas e/ou arquiteturas) possíveis de serem avaliados. São inicialmente selecionados todos os que atendem as restrições já definidas no passo de Análise e Competências do Usuário, os quais também devem ser avaliados por Adequabilidade-Exequibilidade-Aceitabilidade (AEA). Como resultado dessa fase, é gerado o documento de Plano de Avaliação.

A última fase é a **Apresentação dos Resultados**, na qual os resultados dos experimentos conduzidos pelos possíveis fornecedores das novas arquiteturas e/ou conduzidos no sistema atual em uso, são reunidos e analisados. Com a avaliação dos dados obtidos nos testes, é possível determinar a precisão relativa dos sistemas mencionados. Os dados apresentados pelos executores são analisados, aplicando-se os valores coletados versus MEOs. Neste estágio é possível ter um esquema como o que está representado na Figura 2. Vale observar que esse exemplo é o caso mais complexo, pois além da existência das diversas aplicações, essas ainda precisam ser mapeadas para *benchmarks* representativos (as próprias aplicações não serão executadas para os testes).

A partir do *workflow* com as aplicações  $A_h$  onde  $h = 1, \dots, y$  é possível mapear cada uma das aplicações  $A_h$  para as classes  $C_i$ , onde  $i = 1, \dots, l$  as quais serão mapeadas para os *benchmarks*  $B_j$ ,  $j = 1, \dots, n$ . Tem-se então:

$$A_h \implies C_i \implies B_j$$

Seguindo essa relação, os pesos  $w_i$ , inicialmente atribuídos às classes  $C_i$ , são transferidos para os *benchmarks*  $B_j$  resultando em um conjunto de pesos  $w_j$ ,  $j = 1, \dots, n$ , sendo que  $\sum_{j=1}^n w_j = 1$ . Vale observar que, em casos mais simples, em que esse mapeamento não é necessário, os pesos podem ser atribuídos diretamente às aplicações.



Na sequência, para cada equipamento  $E_k$ , ( $k = 1, \dots, m$ ) são obtidos os valores operacionais referentes ao desempenho da execução de cada aplicação (ou *benchmark*) em cada um dos  $m$  equipamentos. Ou seja, os tempos de execução de cada aplicação  $j$  em cada equipamento  $k$ , representada por  $t(j, k)$ . Também serão obtidos os valores operacionais referentes ao custo de cada equipamento  $c_k$ .

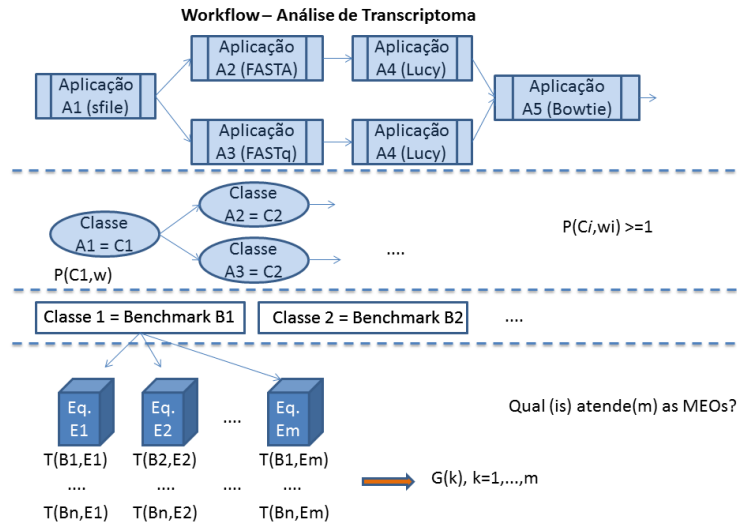


Figura 2: Exemplo de mapeamentos feitos para o *workflow* de aplicações e cálculos para avaliação do desempenho.

### 3.1. Função de Ganho

Com a obtenção dos valores operacionais referentes aos tempos de execução  $t(j, k) =$  unidades de tempo (u.t) e os custos de cada equipamento  $c_k =$  unidades monetárias (u.m) foi desenvolvida uma função de ganho para essa metodologia. A função permite quantificar qual o melhor equipamento em termos de desempenho (tempos de execução) e custo para a execução de um conjunto de aplicações científicas. Ou seja, é possível priorizar o que é mais importante para o usuário, o menor custo do equipamento ou o menor tempo de execução. Além disso, a função também considera os pesos atribuídos a essas aplicações.

Para utilização da função de ganho, define-se o custo relativo do equipamento como  $C_k = \frac{1}{c_k}$  e o tempo de execução de cada equipamento  $k$  para cada aplicação  $j$ , como  $T_{j,k} = \frac{1}{t(j,k)}$ . Dessa forma, o valor do equipamento de maior custo (situação que não é desejável) passa a ter uma parcela menor de contribuição na função que representa o ganho final de cada equipamento. O mesmo se dá com o tempo de execução, que quanto menor (condição desejável), passará a ter uma contribuição maior no valor final da função de ganho.

Esses valores são então normalizados para torná-los adimensionais. A normalização de  $T(j, k)$ , obtendo seu valor normalizado  $D(j, k)$  é dada por:

$$D(j, k) = \frac{T(j,k)}{\sum_{p=1}^m T(j,p)}, \text{ para todo } j = 1, \dots, n \text{ e } k = 1, \dots, m$$

Com a normalização de  $C_k$ , obtém-se  $C_{E_k}$  da seguinte forma:

$$C_{E_k} = \frac{C_k}{\sum_{p=1}^m C_p}, k = 1, \dots, m$$

A partir desses valores normalizados é possível aplicar a função de ganho e avaliar cada equipamento, levando em consideração o conjunto de aplicações que ele vai executar. A atribuição de pesos ao desempenho e ao custo é denotada por  $w_d$  e  $w_c$ , respectivamente. A função de ganho ( $G(k)$ ) é dada por 1:

$$G(k) = (w_d \cdot \sum_{j=1}^n w_j \cdot D(j, k)) + w_c \cdot C_{E_k}, k = 1, \dots, m \quad (1)$$

Essa função foi obtida da seguinte forma:

Atribuindo pesos distintos  $w_c$  e  $w_d$ , para o custo e para o desempenho respectivamente, onde  $w_c$  e  $w_d$  são constantes tais que  $w_c + w_d = 1$ , tem-se que a função de ganho para execução de cada aplicação, em um equipamento, dada por:

$$g(j, k) = w_c \cdot C_{E_k} + w_d \cdot D_{j,k}$$

Após a atribuição dos pesos  $w_j$  para as aplicações (que podem ser os *benchmarks* mapeados a partir das aplicações do *workflow* e das classes), essa função pode ser ampliada para abranger diversas aplicações e diversos equipamentos, da seguinte forma:

$$\begin{aligned} G(k) &= \sum_{j=1}^n (w_j \cdot g(j, k)) \\ &= \sum_{j=1}^n w_j (w_c \cdot C_{E_k} + w_d \cdot D_{j,k}), k = 1, \dots, m \end{aligned}$$

Ou seja,  $G(k)$  representa o ganho do equipamento  $E_k$  ao executar todas as aplicações, considerando os pesos atribuídos a cada aplicação, e também referente a importância relativa do custo e do desempenho.

Como  $w_d$ ,  $w_c$  e  $C_{E_k}$  não dependem de  $j$ , pode-se reescrever a função acima da seguinte forma:

$$G(k) = (w_c \cdot C_{E_k} \cdot \sum_{j=1}^n w_j) + (w_d \cdot \sum_{j=1}^n w_j \cdot D_{j,k})$$

Como  $\sum_{j=1}^n w_j = 1$  obtém-se então a Função 1<sup>3</sup>.

Assim é possível identificar o equipamento que apresenta maior ganho, como  $MAX(G(k))$ , ou seja, o que tiver o máximo valor do ganho  $G(k)$ . Além disso, é possível definir pelas MEOs aqueles que atendem aos requisitos de desempenho exigidos para o conjunto de aplicações científicas do usuário e a utilização dos pesos permite a escolha de um equipamento que atenda a necessidades mais específicas.

<sup>3</sup>Esse formato final da função foi rearranjado dessa forma por ser menos custoso caso seja implementada computacionalmente

As avaliações dos dados versus MEO são apresentados no relatório de análise, mostrando aqueles que não atendem e os que atendem às medidas de desempenho, com seus respectivos custos de execução para aquisição, garantias, etc. No caso da concepção dos experimentos se mostrarem inadequadas, é necessário voltar a fase 2, identificar e corrigir quais aspectos do problema não foram detalhados adequadamente. Além disso, se a concepção parecer válida mas os dados coletados forem considerados insuficientes para avaliar todos os propósitos determinados, torna-se necessário reprogramar os experimentos, retornando a Fase 3. Com a apresentação do *Relatório Final da Análise* encerra-se a fase inicial da AO e inicia-se a AO Continuada.

Cada um desses passos são descritos e detalhados nos seus aspectos conceituais e como realizar sua aplicação em [18].

#### 4. ESTUDO DE CASO

Para o desenvolvimento da metodologia foram realizados dois conjuntos experimentais e um estudo de caso. Os experimentos foram conduzidos em arquiteturas de alto desempenho e focados em aplicações científicas paralelas de memória compartilhada. Porém, devido às limitações de espaço, neste trabalho será apresentado apenas o estudo de caso. O conjunto completo de experimentos encontra-se em [18].

A metodologia foi avaliada em um estudo de caso na bioinformática para o sequenciamento do transcrito de bactérias importantes para o controle de infecções hospitalares e doenças como a tuberculose. Com o objetivo de viabilizar a análise dessas sequências com maior precisão e tempo viável é que se decidiu pela aplicação da metodologia desenvolvida neste trabalho e assim determinar a arquitetura de alto desempenho mais adequada para esse problema. O foco da AO foi a execução de três aplicações: o BLAST (*Basic Local Alignment Search Tool*), o MUMmer e o K-means, em tempo aceitável de execução.

Atualmente essas aplicações são executadas no pipeline tradicional de trabalho, usando sequências de referência e as próprias estações de trabalho dos pesquisadores. Assim, o objetivo é utilizar um novo pipeline, no qual as sequências de referência não serão utilizadas. A restrição é que o novo equipamento que venha a ser adquirido execute o BLAST em até dez dias, para os tamanhos de problema que estão envolvidos neste tipo de pesquisa. Assim, o processo de aquisição de um novo equipamento foi motivado pela necessidade de execução desse novo pipeline de trabalho, os quais exigem recursos computacionais com maior capacidade.

Neste trabalho, na **seleção das arquiteturas de teste**, segundo os critérios de AEA, haviam duas arquiteturas adequadas, apresentadas na Tabela 1. Além disso, os custos para cada arquitetura foram obtidos junto aos fornecedores e apresentados na Tabela 2 com valores normalizados, conforme as definições utilizadas na Função Ganho.

As avaliações foram feitas usando os valores operacionais (tempos de execução e custos) e a importância relativa entre as aplicações e entre custo e desempenho. As aplicações foram executadas nas arquiteturas A e B e os tempos de execução obtidos são apresentados na Tabela 3. Estes valores operacionais são apresentados segundo a nomenclatura utilizada na definição da função de Ganho. Na Tabela 3, na primeira coluna é apresentado para cada experimento ( $t(j, k)$ ), para cada aplicação  $j$ , executada em cada arquitetura  $k$  (Tabela 1) seguido pelos tempos médios de execução em segundos (segunda

	Arq A	Arq B
Pico Teórico	1177	563,2
Desempenho	GFlops	GFlops
Memoria Bandwidth	51.2 GB/s	51.2 GB/s
Memory Tam	128 GB	64 GB
Núcleos	64	32
Clock (GHz)	2.3	2.2

Tabela 1: Arquiteturas computacionais avaliadas.

$c(k)$	Valor (\$)	$CE(k)$
$c(A)$	8900	0.00011
$c(B)$	8760	0.00011

Tabela 2: Valores operacionais para os custos de cada arquitetura e seus valores após a normalização.

coluna - tempo de execução (s)).

$t(j, k)$	Tempo Exec. (s)
$t(blast, A)$	79341
$t(blast, B)$	193515
$t(kmeans, A)$	143
$t(kmeans, B)$	121
$t(mum, A)$	42
$t(mum, B)$	38

Tabela 3: Tempo médio de execução para Blast, K-means e MUMmer ( $j = blast, kmeans, mum$ ).

Nas avaliações apresentadas na Tabela 4, os ganhos foram avaliados usando pesos iguais para custo e desempenho ( $w_c$  e  $w_d$ ) já que os pesquisadores não manifestam preferências entre eles. Na primeira coluna, os pesos entre as aplicações também são iguais e os ganhos foram avaliados sem preferências. Na segunda coluna, os pesos representam a importância das aplicações para a pesquisa e foram definidos de forma verbal e subjetiva pelos pesquisadores. Há uma preferência absoluta pela aplicação do Blast em relação ao MUMmer e K-means e uma preferência moderada pelo MUMmer em relação ao K-means. Os valores numéricos dos pesos foram estabelecidos usando o método AHP [?].

Se olharmos para os valores operacionais (tempos de execução -  $t(j, k)$  - Tabela 3) é possível notar que a aplicação Blast tem um melhor desempenho na arquitetura A, enquanto MUMmer e K-means têm o menor tempo de execução na arquitetura B. Nessa situação real, ao se avaliar diferentes aplicações científicas, os resultados são divergentes em termos da melhor arquitetura para sua execução. Como visto nos exemplos, os custos são muito semelhantes e a decisão sobre qual é a melhor arquitetura para atender as três aplicações não é simples. Assim, a tomada de decisão baseou-se nos valores quantitativos obtidos com o tempo de execução das aplicações e custos das arquiteturas, somado das características qualitativas e dos resultados finais obtidos com a função de Ganho. Quando os pesos são iguais, o maior ganho é apresentado pela arquitetura A, apesar da

<i>Pesos</i>		
$w_{blast}$	0.333	0.794
$w_{kmeans}$	0.333	0.067
$w_{mum}$	0.333	0.140
$w_c$	0.5	0.5
$w_d$	0.5	0.5
<i>Ganho(k)</i>		
$Ganho(A)$	0.5190	0.5782
$Ganho(B)$	0.4760	0.4223

Tabela 4: Definição dos pesos e os ganhos para cada arquitetura.

pequena diferença. Porém, quando a importância relativa atribuída pelos pesquisadores foi considerada, o ganho da arquitetura A é ainda maior.

Assim, com base nesses resultados finais utilizando a Função de Ganho, foi possível uma tomada de decisão confiável sobre a arquitetura de HPC que traria o maior ganho para as pesquisas, ao executar os experimentos com as aplicações avaliadas.

O estudo de caso foi focado na avaliação, principalmente da fase final da AO, e possibilitou a validação da função de ganho e da utilidade da metodologia e das suas contribuições para os pesquisadores de outras áreas, como a bioinformática.

## 5. CONSIDERAÇÕES FINAIS

Apesar dos sistemas de HPDC serem essenciais para alavancar o progresso científico em muitos domínios de aplicação e a sua disponibilidade até já é considerada fator de soberania nacional em alguns países, os métodos para aquisição e avaliação destes equipamentos ainda são limitados. É evidente que é preciso evoluir os procedimentos de avaliação e testes, considerando o conjunto de aplicações e equipamentos, testando assim o sistema de HPDC em operação em ambiente tão real quanto possível, determinando o quanto bem ele pode cumprir seu objetivo, que é a execução de um conjunto de aplicações científicas, segundo o foco deste trabalho.

Assim, este trabalho se dedicou a criar uma metodologia que oriente o pesquisador/técnico por essa tomada de decisão, permitindo a avaliação de equipamentos de HPDC com foco nos requisitos das suas aplicações científicas. Apesar do sucesso da AO em áreas como a militar, governamental e industrial, ainda não havia uma metodologia de AO voltada para a aquisição e manutenção de sistemas de HPDC. A metodologia desenvolvida compreende uma sistemática de avaliação que se inicia com uma definição clara a respeito do problema a ser resolvido pela AO e segue por diversas etapas de análise. Essas etapas são divididas em uma sequência de fases e passos os quais resultam em relatórios que consolidam e documentam cada análise realizada. Em alguns desses passos são previstas e guiadas a execução de testes controlados, em ambientes tão reais quanto possível, os quais permitem a análise final dos resultados usando uma função de ganho, desenvolvida especificamente para essa metodologia.

Os resultados finais permitem medir qualitativa e quantitativamente os ganhos de cada equipamento avaliado e tomar a decisão com base nestas medidas. Ainda, após

a aquisição do equipamento, a metodologia guia o pesquisador por avaliações periódicas, por meio da AO continuada.

A metodologia foi avaliada em um estudo de caso para a área da bioinformática. O problema abordado no estudo de caso limitou-se a duas aplicações e não envolveu a parte da metodologia referente à fornecedores externos e nem da avaliação da parte física que receberia a possível infraestrutura computacional. Entretanto, as contribuições com a aplicação da metodologia se mostraram relevantes, pois a decisão sobre a aquisição de qual a melhor arquitetura para atender as duas aplicações, com os requisitos estabelecidos pelos pesquisadores, mostrou-se complexa.

A principal contribuição deste trabalho, é o desenvolvimento da metodologia para avaliação e manutenção de equipamentos de HPDC. Esse tipo de metodologia, ainda não existe no Brasil e mesmo em outros países está restrita aos grandes centros de computação, os quais desenvolvem trabalhos para área governamental e de defesa. Porém, suas metodologias são proprietárias, desenvolvidas especificamente ao seu grupo de aplicações científicas e com o conhecimento adquirido ao longo de anos de trabalho e pesquisa. Além disso, todo conhecimento necessário para se desenvolver a base de uma metodologia como esta também não é divulgada.

Assim, juntamente com os benefícios financeiros e operacionais que podem ser obtidos, a divulgação dessa metodologia pode trazer uma mudança de paradigma de como os sistemas computacionais podem ser avaliados, pois seu desenvolvimento também resultou em um manual de emprego que guia o usuário na sua aplicação e, com foco nessa mudança de paradigma, encontra-se disponível *on-line* na biblioteca do LNCC [19].

Apesar das contribuições já obtidas com o desenvolvimento do trabalho, os quais resultaram nesta metodologia, o estudo pode ser dito inicial se considerado a extensão do problema a ser avaliado e as inúmeras avaliações que ainda podem ser feitas.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] STEVENS, R. T. *Operational Test and Evaluation: A Systems Engineering Process*. Malabar, Florida: Robert E. Krieger Publishing Company, 1986. 2
- [2] JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: A review. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 31, n. 3, p. 264–323, 1999. ISSN 0360-0300. Disponível em: <<http://doi.acm.org/10.1145/331499.331504>>. 2
- [3] DANTAS, M. *Computação Distribuída de Alto Desempenho: redes, grids e clusters computacionais*. [S.l.]: Axcel Books, 2005. 207–208 p. 3
- [4] HEY, T.; TANSLEY, S.; TOLLE, K. (Ed.). *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Redmond, Washington: Microsoft Research, 2009. Disponível em: <<http://research.microsoft.com/en-us/collaboration/fourthparadigm/>>. 3
- [5] JAIN, R. *The Art of Computer Systems Performance Analysis - Techniques for experimental design, measurement, simulation, and modeling*. [S.l.]: Wiley, 1991. I-XXVII, 1-685 p. (Wiley Professional Computing). ISBN 978-0-471-50336-1. 4. 5

- [6] COLELLA, P. Defining software requirements for scientific computing. DARPA HPCS presentation. 2004. 4
- [7] ASANOVIC, K. et al. A View of the Parallel Computing Landscape. *Commun. ACM*, ACM, New York, NY, USA, v. 52, n. 10, p. 56–67, oct 2009. ISSN 0001-0782. Disponível em: <<http://doi.acm.org/10.1145/1562764.1562783>>. 4
- [8] PATTERSON, D. Orgins and Vision of the UC Berkeley Parallel Computing Laboratory. In: \_\_\_\_\_. 1. ed. [S.l.]: Microsoft Corporation, 2013. cap. 1, p. 11–42. 4
- [9] CHOUDHARY, A. et al. *The DAMSEL Project - A Data Model Storage Library for Exascale Science*. 2013. [Http://science.energy.gov/damsel-110221.pdf](http://science.energy.gov/damsel-110221.pdf). 4
- [10] JOHN, L. K.; EECKHOUT, L. (Ed.). *Performance Evaluation and Benchmarking*. [S.l.]: CRC Press, 2006. 5
- [11] BUZEN, J. P. Fundamental operational laws of computer system performance. *Acta Informatica*, Springer Berlin / Heidelberg, v. 7, p. 167–182, 1976. ISSN 0001-5903. 10.1007/BF00265769. Disponível em: <<http://dx.doi.org/10.1007/BF00265769>>. 5
- [12] DENNING, P. J.; BUZEN, J. P. The operational analysis of queueing network models. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 10, n. 3, p. 225–261, set. 1978. ISSN 0360-0300. Disponível em: <<http://doi.acm.org/10.1145/356733.356735>>. 5
- [13] BUZEN, J. P. Operational Derivations of some Basic Results in Queueing Theory. In: BOKSENBAUM, M. et al. (Ed.). *Int. CMG Conference*. [S.l.]: Computer Measurement Group, 1982. p. 356–361. 5
- [14] BUZEN, J. P. Queueing Models: Beyond the Basics. In: MORLEY, D. et al. (Ed.). *Twelfth International Computer Measurement Group Conference, Las Vegas, NV, USA, December 8-12, 1986, Proceedings*. [S.l.: s.n.], 1986. p. 727–732. 5
- [15] KELLY, T. et al. Operational analysis of parallel servers. In: MILLER, E. L.; WILLIAMSON, C. L. (Ed.). *MASCOTS*. [S.l.]: IEEE Computer Society, 2008. p. 227–236. ISBN 978-1-4244-2818-2. 5
- [16] SHEN, K. et al. Operational analysis of processor speed scaling. In: *Proceedings of the twentieth annual symposium on Parallelism in algorithms and architectures*. New York, NY, USA: ACM, 2008. (SPAA '08), p. 179–181. ISBN 978-1-59593-973-9. Disponível em: <<http://doi.acm.org/10.1145/1378533.1378566>>. 5
- [17] BUZEN, J. P. Ubiquity symposium: Computation, Uncertainty and Risk. *Ubiquity*, ACM, New York, NY, USA, v. 2011, jan 2011. ISSN 1530-2180. Disponível em: <<http://doi.acm.org/10.1145/1922681.1936886>>. 5
- [18] FERRO, M. *Avaliação de Sistemas de Computação Massivamente Paralela e Distribuída: Uma metodologia voltada aos requisitos das aplicações científicas*. Tese (Tese de Doutorado) — Laboratório Nacional de Computação Científica, Petrópolis - RJ, Maio 2015. 8, 11
- [19] FERRO, M.; MURY, A. R.; SCHULZE, B. *Manual de Metodologia de Análise Operacional de Sistemas de Computação Científica Distribuída de Alto Desempenho*. Petrópolis - RJ, 2015. Disponível em: <[www.lncc.br](http://www.lncc.br)>. 14