

# Assistente Interativo para Varredura Paramétrica em Aplicações de Alto Desempenho ¶

Eric B. Gauch<sup>1,3</sup>, Vitor H. L. Mesquita<sup>1,2</sup>, Bruno Silva<sup>1</sup>, Marco A. S. Netto<sup>1</sup>

<sup>1</sup>IBM Research Brasil  
Rua Tutóia, 1157 - São Paulo - SP - Brasil

<sup>2</sup>Graduando do Curso de Análise e Desenvolvimento na FATEC-SP  
Praça Coronel Fernando Prestes, 30 - São Paulo - SP - Brasil

<sup>3</sup>Graduando do Curso de Ciência da Computação na Universidade Presbiteriana Mackenzie  
Rua da Consolação, 930 - Consolação, São Paulo - SP - Brasil

{ebgauch,vleal,sbruno,mstelmar}@br.ibm.com

**Abstract.** *Parameter sweep applications are used when it is necessary to find a set of parameters that optimizes a certain process. Often High Performance Computing is used as a resource to speed up the search of different groups of parameters, mainly in cases where the execution of many scenarios may require prohibitive amounts of computational resource and time. This work presents an overview of a tool to accelerate the process of finding the ideal set of parameters in a system. For such, the human-in-the-loop interaction, optimization algorithms, and visualizations are used to reach the expected result with the least possible resources.*

**Resumo.** *Aplicações paramétricas são utilizadas quando é necessário encontrar um conjunto de parâmetros que otimize um determinado processo. Frequentemente a Computação de Alto Desempenho ou High Performance Computing (HPC) é utilizada como recurso para acelerar a busca de diferentes grupos de parâmetros, principalmente em casos onde a execução de muitos cenários pode demandar quantidades de recursos computacionais e tempo proibitivos. Este trabalho apresenta uma visão geral de uma ferramenta para acelerar o processo de busca dos parâmetros ideais em um sistema. Para tal, a interação do humano no processo, algoritmos de otimização e visualizações são utilizados para atingir o resultado esperado com o mínimo de recurso possível.*

## 1. Introdução

A avaliação de cenários de teste e a calibração de modelos são comuns em diversas indústrias, incluindo financeira, aeroespacial, saúde e energética. Normalmente, os usuários executam aplicações que contêm um conjunto de parâmetros, cada um capaz de assumir um ou mais intervalos de valores. Estas aplicações são conhecidas como aplicações de varredura de parâmetros ou aplicações paramétricas. Nesse contexto, é necessário executar várias instâncias da aplicação com os valores de seus parâmetros para encontrar uma combinação que atende a uma determinada especificação ou critérios de otimização. Muitas dessas aplicações precisam ser executadas em computadores de grande porte devido

---

¶Este trabalho é apoiado e parcialmente custeado pela FINEP / MCTI, contrato nº 03.14.0062.00.

não apenas a grande quantidade de execuções mas também a complexidade computacional de cada uma delas.

Nos últimos anos, muitos algoritmos de otimização combinatória têm sido propostos. Contudo, soluções totalmente automáticas também apresentam alguns problemas como por exemplo: (i) mesmo quando o modelo de otimização representa bem o problema em questão, muitas vezes o método aplicado pode demorar para encontrar uma solução e violar restrições de tempo e/ou custo; e (ii) a experiência e criatividade de especialistas na área em questão são difíceis de se representar principalmente em problemas que envolvem decisões complexas e importantes (exemplo, sistemas de suporte para compra e venda de ações).

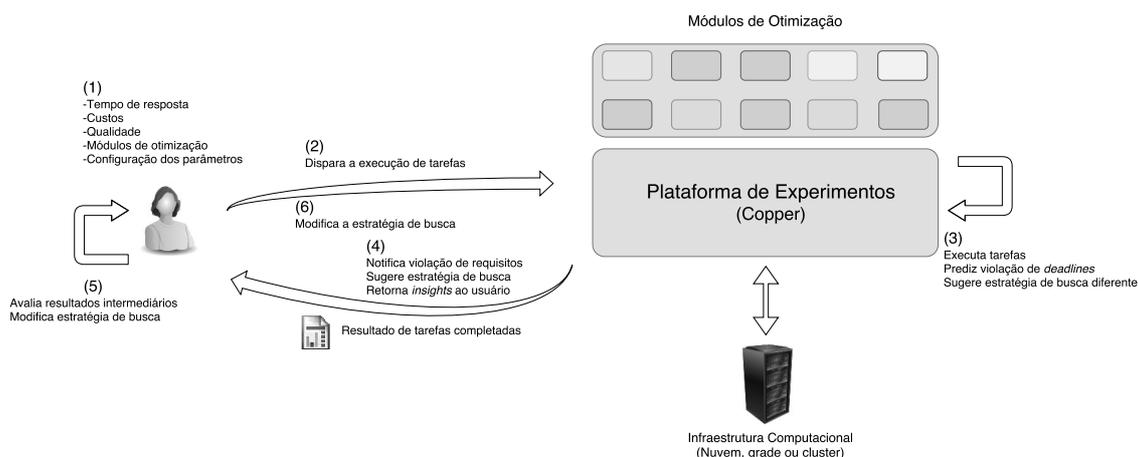
No intuito de resolver as questões acima, a ferramenta *Cognitive Optimizer and Parameter Explorer* (Copper) foi proposta para auxiliar analistas na busca de soluções otimizadas dentro de um espaço de amostras discretas. A ferramenta combina conhecimento e experiência humana ao lado de técnicas de otimização combinatória para alcançar essas soluções. Com o Copper, os usuários podem descobrir quais são os parâmetros mais importantes em sua aplicação e usar sua experiência para diminuir o tempo de avaliação do sistema. Este trabalho apresenta uma visão geral da arquitetura da ferramenta Copper e como seus módulos, aliados ao conhecimento humano, impactam no resultado final conforme descrito por Silva *et al.* (2016).

## 2. Trabalhos Relacionados

Parker & Johnson (1995) apresentaram um sistema chamado SCIRun, descrito como um modelo de programação de fluxo de dados e programação visual para simplificar tarefas de criação, depuração, otimização e controle de simulações científicas complexas. Meignan *et al.* (2015) exploraram os diferentes papéis que um usuário pode ter em um processo de otimização, como ajustar ou adicionar novas restrições e objetivos, ajudando e guiando o próprio processo de otimização, fornecendo informações relacionadas a variáveis de decisão. Os autores do Nascimento & Eades (2005) propuseram uma aplicação para auxiliar os usuários na assistência do processo de otimização através do conhecimento do domínio aplicado, reduzindo o espaço de busca a ser explorado, e evitando ambiguidade em múltiplas soluções. Abramson *et al.* (2011) desenvolveram um sistema para facilitar a exploração dos parâmetros e suporte para a abstração da plataforma de computação subjacente.

## 3. Copper

A Figura 1 apresenta uma visão geral da arquitetura da ferramenta. Visando facilitar otimizações de múltiplos objetivos, a ferramenta dispõe de módulos (diferentes técnicas de análise e otimização dos dados) onde o usuário poderá escolher qual a abordagem desejada. Dois dos módulos utilizados são: “Análise de Preferências” que analisa o comportamento do usuário e o “Provedor de Dicas” que provê possíveis *insights* durante a execução. Inicialmente o analista faz a seleção dos valores de cada parâmetro, apresenta as restrições de custo, tempo, qualidade do experimento e objetivos de otimização (1). Em seguida, o usuário dispara a execução de um conjunto de tarefas de avaliação, onde o sistema utiliza o conceito de *bag of tasks* (2). Posteriormente o Assistente de Busca executa as tarefas, prediz a violação de *deadlines* e sugere mudanças na estratégia de busca baseada



**Figura 1. Visão de alto nível do fluxo de informação da ferramenta Copper.**

na estratégia *Greedy Randomized Adaptive Search Procedure* (GRASP) Feo & Resende (1995) (3). Em seguida, o usuário é notificado e o resultado das tarefas é apresentado (4). O analista verifica os resultados e modifica a estratégia de busca (se necessário) (5). Por fim, o usuário notifica o sistema sobre a mudança na estratégia (6).

O Copper está sendo utilizado para maximização de produção agrícola por meio do ambiente de simulação PCSE (Python Crop Simulation Environment) para simular a produção de trigo de inverno para a região de Anhui na China. Os parâmetros do modelo são as quantidades de irrigação/fertilizante aplicados no plantio. Nos estudos conduzidos, foram percebidos que a quantidade de fertilizante aplicados no período intermediário da colheita afetam fortemente a produção.

Outro estudo de caso realizado é o *Imgexp (Image experiment framework)*, que utiliza processamento de imagens sísmicas e técnicas de aprendizado de máquina para identificar diferentes texturas na imagem correspondentes às camadas do solo marítimo, como por exemplo sal e pré-sal.

A Figura 2 apresenta a tela principal da interface gráfica da ferramenta Copper. A interface é baseada em tecnologia WEB e o *framework* D3 é utilizado para geração dos gráficos. É possível observar os diferentes módulos acoplados ao *Cockpit*. Conforme as tarefas são executadas, o usuário recebe diferentes tipos de informações que contemplam os parâmetros, resultados (*output*), custos/tempo, sugestões e dicas.

#### 4. Conclusão

Diversas áreas em ciência e engenharia precisam de várias execuções da mesma aplicação com diferentes parâmetros. Essas execuções são responsáveis pela avaliação de processos complexos e calibração de modelos físicos. A execução de todos os valores possíveis para todos os parâmetros suportados nem sempre é possível dadas restrições de tempo, custo ou energia. Assim, ferramentas de apoio a avaliação de sistemas com múltiplos parâmetros é de extrema importância em ambientes de processamento de alto desempenho.

A ideia de acoplar módulos adaptáveis a situações distintas e aplicações específicas é um dos diferenciais do Copper. Para o usuário realizar grandes experimentos computacionais, esperamos que funcionalidades de visualização de resultados, auxílio nas decisões de

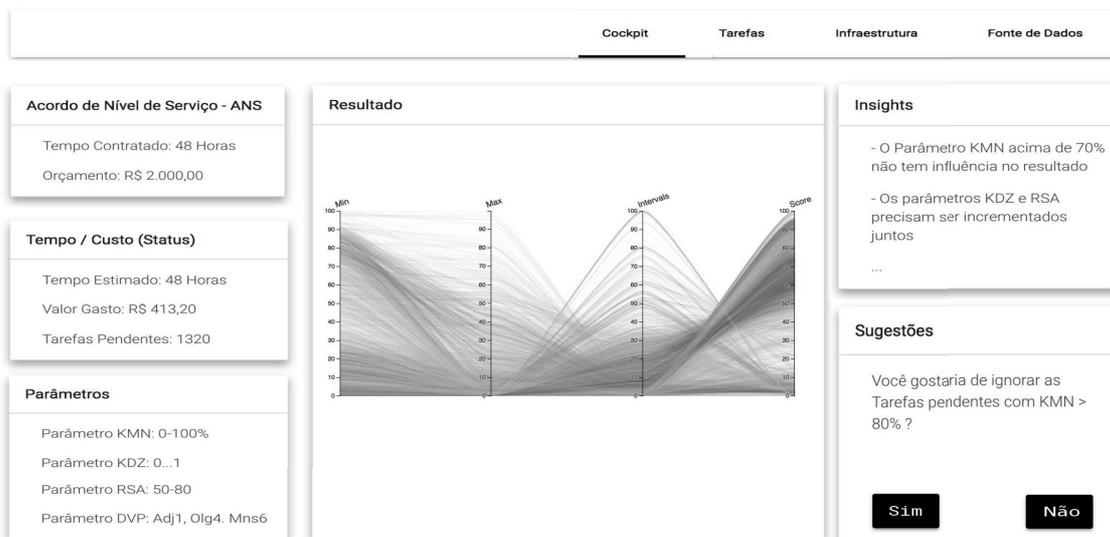


Figura 2. Exemplo de interface gráfica Copper.

otimização e dicas para descartar tarefas e priorizá-las, podem fazer com que os usuários sejam mais produtivos e possam trazer benefícios tanto para ciência quanto para negócios de maneira mais eficiente.

## Referências

- Abramson, David, Bethwaite, Blair, Enticott, Colin, Garic, Slavisa, & Peachey, Tom. 2011. Parameter exploration in science and engineering using many-task computing. *Parallel and Distributed Systems, IEEE Transactions on*, **22**(6), 960–973.
- do Nascimento, Hugo AD, & Eades, Peter. 2005. User hints: a framework for interactive optimization. *Future Generation Computer Systems*, **21**(7), 1177–1191.
- Feo, Thomas A, & Resende, Mauricio GC. 1995. Greedy randomized adaptive search procedures. *Journal of global optimization*, **6**(2), 109–133.
- Meignan, David, Knust, Sigrid, Frayret, Jean-Marc, Pesant, Gilles, & Gaud, Nicolas. 2015. A Review and Taxonomy of Interactive Optimization Methods in Operations Research. *ACM Transactions on Interactive Intelligent Systems (TiIS)*, **5**(3), 17.
- Parker, Steven G, & Johnson, Christopher R. 1995. SCIRun: a scientific programming environment for computational steering. *In: Proceedings of the 1995 ACM/IEEE conference on Supercomputing*. ACM.
- Silva, Bruno, Netto, Marco AS, & Cunha, Renato LF. 2016. SLA-aware Interactive Workflow Assistant for HPC Parameter Sweeping Experiments. *In: Proceedings of the Int. Workshop on Workflows in Support of Large-Scale Science in conjunction with Int. Conf. for High Performance Computing, Networking, Storage and Analysis (WORKS@SC)*.