

1 Introdução

1.1 Motivação

O investimento em infraestrutura computacional para suportar picos de processamento de curta duração ou sazonais pode gerar desperdícios financeiros, em razão de na maior parte do tempo estes recursos fiquem ociosos. Além disso, em muitas soluções o tempo de resposta é crítico para atendimento dos requisitos do negócio, sendo necessário um alto investimento em infraestrutura, tornando, muitas vezes, a solução economicamente inviável. Neste cenário é fundamental a alocação inteligente de recursos computacionais em função da demanda por processamento, custo desta alocação e requisitos do negócio.

A Simulação de Monte Carlo é um método estatístico utilizado para resolver uma gama de problemas científicos e de engenharia. Quando aplicado a problemas reais requer robusta infraestrutura computacional, o que muitas vezes impede sua aplicação em níveis confiáveis. Por outro lado, tal método apresenta características que permitem a adequada paralelização de sua execução e claras situações de pico de processamento. Dessa forma, sem dúvida, existem diversos exemplos típicos do cenário acima proposto.

Computação na nuvem surge como uma alternativa para disponibilizar recursos computacionais sob demanda, gerando economia de escala sem precedentes e escalabilidade quase infinita.

Com uma moderna arquitetura de software é possível desenvolver blocos de código que podem ser facilmente compartilhados entre diversas aplicações, centralizando assim, sua manutenção, evolução e infraestrutura, minimizando custos e potencializando seu reuso. Ao alinhar uma moderna arquitetura a nuvem é possível encapsular funcionalidades e oferecer um leque de serviços que antes seriam restritos a domínios específicos.

Neste trabalho construímos um arcabouço genérico, a fim de disponibilizar um leque de serviços baseados na Simulação de Monte Carlo. Acreditamos que esta arquitetura será bastante útil em uma rica gama de domínios, ao mesmo tempo fará uso racional da elasticidade provida pela nuvem de modo a alcançar melhores patamares de eficiência e reuso.

1.2 Principais Contribuições

As principais contribuições deste trabalho são:

- Arcabouço genérico para desenvolvimento de leques de serviços baseados na Simulação de Monte Carlo em ambiente de computação na nuvem;
- Algoritmo para definir a melhor configuração do ambiente de computação na nuvem para execução da simulação, que considera características genéricas, mas pode ser adaptado para domínios específicos;
- Prova de conceito do arcabouço, através de sua implantação em um problema de fácil entendimento, com intuito de comparar sua execução com a computação tradicional;
- Demonstração da interoperabilidade do arcabouço com aplicações em outras tecnologias;
- Estudo de caso da área de engenharia mecânica, de razoável complexidade, com a finalidade de comprovar viabilidade, eficiência e eficácia do arcabouço proposto;
- Extensão do arcabouço para suporte a simulações desenvolvidas em MATLAB, uma das principais ferramentas de computação científica do mercado, e em C# .Net, que possui sintaxe similar ao C, que muitos pesquisadores dominam.
- Os profissionais que utilizam a Simulação de Monte Carlo, normalmente, não dominam técnicas avançadas de Engenharia de Software e vice-versa. O arcabouço aproxima profissionais dessas duas áreas.

1.3 Organização do Trabalho

O presente trabalho se encontra estruturado em 12 capítulos, a saber:

- **Introdução (Capítulo 1):** Apresentamos de forma concisa a motivação, contribuições e desafios do presente trabalho, assim como, a evolução histórica do tema, a organização do documento e o público alvo do trabalho.
- **Computação na Nuvem (Capítulo 2):** Discorremos sobre o paradigma de computação na nuvem, suas características e seus benefícios.

- **Método de Monte Carlo (Capítulo 3):** Contextualizamos e exemplificamos o Método de Monte Carlo, e concluímos propondo novas perspectivas para o método com o uso de ambientes de computação na nuvem.

- **Solução Proposta (Capítulo 4):** Definimos o escopo de atuação da solução proposta (McCloud), formalizamos sua identidade e apresentamos sua arquitetura conceitual. Além disso, fornecemos uma visão geral da plataforma escolhida e dos motivos desta escolha.

- **Modelo de Programação (Capítulo 5):** Documentamos o modelo de programação adotado na solução proposta, com suas etapas, características e fluxos, demonstramos assim, como é realizado o paralelismo.

- **O Arcabouço de Serviços (Capítulo 6):** A solução proposta do ponto de vista de engenharia de software foi pensada de forma extensível (arcabouço) e interoperável (serviço), sendo apresentado nesse capítulo como fazer uso de seus recursos (pontos de extensão, métodos, instrumentação e etc.) para desenvolver serviços de “simular” sem complexidades adicionais pelo ambiente de computação na nuvem.

- **Prova de Conceito (Capítulo 7):** Diante da solução proposta escolhemos um problema de fácil entendimento, aproximação do número π , para provar que com sua utilização e de ambientes de computação na nuvem é possível realizar simulações com significativo ganho de tempo e custo proporcional aos requisitos da simulação.

- **Duração x Custo (Capítulo 8):** Provamos no capítulo anterior os benefícios da solução proposta, mas estes ganhos são limitados pelo custo de gerenciamento do paralelismo. Nesse capítulo propomos um modelo que permite prever a duração e o custo de uma simulação, e também, considerar as preferências do usuário na relação de duração e custo, para escolha do número de instâncias ótimo.

- **Estudo de Caso (Capítulo 9):** Comprovado os benefícios da solução proposta fomos em busca de uma pesquisa científica em andamento para mensurar na prática o interesse dos usuários na utilização do McCloud e entender os desafios envolvidos em sua adoção. Esse capítulo relata esse estudo de caso da área de Engenharia Mecânica e com suporte a código na linguagem MATLAB.

- **Guia do Usuário (Capítulo 10):** Disponibilizamos nesse capítulo um passo a passo que ensina como usar as implementações disponibilizadas nesse

trabalho, ou seja, com código textual na linguagem C# ou código compilado na linguagem MATLAB, em pesquisas científicas ou comerciais, sem qualquer requisito de conhecimento prévio sobre o ambiente de computação na nuvem, o McCloud ou engenharia de software.

- **Conclusão (Capítulo 11):** Neste capítulo revisitamos as descobertas e analisamos as lições aprendidas, limitações e perspectivas futuras do trabalho.

- **Bibliografia (Capítulo 12):** Coletânea de artigos, livros e sites relevantes aos assuntos tratados no âmbito deste trabalho.

1.4 Público Alvo

O presente trabalho se destina a três públicos distintos:

1) Usuário: Estatísticos, Atuários, Matemáticos e Engenheiros que desejam utilizar uma de suas implementações para incrementar significativamente a precisão de suas aproximações científicas ou comerciais com o Método de Monte Carlo. Nesse trabalho disponibilizamos duas implementações, uma na qual simulações precisam ser modeladas (escritas) com código textual na linguagem C# (Capítulo 7) e outra com código compilado na linguagem MATLAB (Capítulo 9).

2) Desenvolvedor: Profissionais de informática que sabem programar, independente de experiência no ambiente de computação na nuvem, que desejam implementar um serviço que realize simulações de Monte Carlo na nuvem, sem precisar se preocupar com detalhes desse novo ambiente. A exigência de domínio em programação deste desenvolvedor varia com a complexidade do domínio e tecnologias.

3) Especialistas: Profissionais com experiência avançada em programação, em engenharia de software e em ambiente de computação na nuvem com objetivo de iniciar seu projeto a partir de um sólido alicerce. Sua utilização é interessante, por exemplo, em problemas com grande computação (*big-compute*) ou no estudo dos novos desafios da nuvem, i.g. tolerância à falha, estudo da duração x custos no paralelismo, segurança da informação, dentre outros.

Cabe atentar que a leitura para Desenvolvedores e Especialistas deste documento deve ser feita na íntegra, visto que diversos detalhes e dicas técnicas são apresentados de forma distribuída no texto.

O Capítulo 10 apresenta um guia destinado principalmente aos Usuários. Esse público pode lê-lo de forma isolada, sem a compreensão de maiores detalhes do trabalho, para uso exclusivo das implementações disponibilizadas. No entanto, a leitura dos capítulos 1, 2, 3, fornece uma visão introdutória que pode também ser interessante para este público, assim como, os capítulos 7 e 9 que apresentam detalhes de experimentos com as implementações em questão.

1.5

Desafios Enfrentados

Esse trabalho venceu obstáculos significativos, dentre os quais se destacam:

- O desenvolvimento de aplicativos de software para a nuvem ainda é complexo, pois são necessários programadores qualificados, com experiência em testes e monitoramento da execução do programa. Isto porque, os programadores não controlam totalmente o ambiente em que suas aplicações vão executar. Neste cenário se torna fundamental diferenciar entre o comportamento do código e as respostas da infraestrutura. O arcabouço proposto abstrai boa parte desta complexidade aos seus usuários, provendo instrumentação, controle dos fluxos e outras facilidades.
- Ao contrário das plataformas de programação, ainda existem poucos componentes, estruturas de dados e códigos sancionados, disponíveis para reutilização no desenvolvimento de softwares para a nuvem. Esse trabalho incorpora uma valiosa coletânea de referências.
- A estratégia de paralelização precisa ser pensada com muito cuidado a fim de evitar que seus custos de gestão e armazenamento não sobreponham os ganhos de eficiência no processamento, sempre mantendo os resultados corretos. Essa questão foi considerada em todo o arcabouço e abordada com profundidade em um capítulo.

1.6 Evolução do Tema

A utilização de computação paralela, com a finalidade de viabilizar a aplicação do Método de Monte Carlo em casos de processamentos intensivos, vem sendo sugerida, implementada e utilizada de diferentes formas nos últimos 20 anos, fazendo uso exclusivamente de *clusters* [67] e, mais recentemente, de computação em grade [68].

Em 25 de junho de 2010, em um episódio do “channel9.msdn.com” da Microsoft, canal onde ideias técnicas e tecnologias são difundidas aos programadores de todo o mundo, foi apresentada uma implementação denominada FullMonte [66], desenvolvida pelo grupo de difusão de tecnologias da empresa, conhecido como “Evangelistas”. O intuito desta implementação foi demonstrar a utilização da fila e das instâncias do Azure para paralelizar o processamento, tendo a mesma sido descontinuada.

Sem conhecimento do FullMonte, nesta mesma época, em disciplina de computação na nuvem da PUC-Rio, realizamos os primeiros experimentos com Monte Carlo em ambiente de computação na nuvem e apresentamos a primeira versão do McCloud, que resultou, no final deste mesmo ano, na primeira submissão de artigo sobre o tema. No final de 2010, quando tomamos conhecimento do FullMonte, as soluções estavam niveladas em conclusões e avanços.

A versão atual do McCloud representa um grande avanço em relação às propostas anteriores, pois este se trata de um arcabouço de serviços e não apenas uma implementação específica como em todos os casos anteriores.

O McCloud, portanto, é a única solução identificada que permite executar qualquer simulação de Monte Carlo na nuvem, através da implementação dos seus pontos de extensão. Os benefícios da ida para nuvem são explorados ao longo dos próximos capítulos.

Recentemente um artigo sobre a última versão do McCloud foi publicada na “Cloud Futures 2012” e seu código fonte foi disponibilizado em repositório público junto a informações adicionais da solução (<http://mccloud.codeplex.com>).