

# Tratamento de Valores Ausentes na Alocação de Máquinas Virtuais para a Computação em Nuvem\*

Marcos Ricardo S. Oliveira<sup>1</sup>, Maycon L. M. Peixoto<sup>1</sup>,  
Fabiola G. P. Greve<sup>1</sup>, Ricardo A. Rios<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)  
Av. Adhemar de Barros, s/n - Campus de Ondina  
Salvador - Bahia - Brasil - CEP 40.170-110

{mroliveira, mayconleone, fabiola, ricardorios}@dcc.ufba.br

**Abstract.** *Virtualization mechanisms are frequently adopted in cloud computing to, transparently and dynamically, make resources available to users. The configuration of such mechanisms relies on information about provided services as, for example, by monitoring workload, one can estimate the number of virtual machines and processors required to meet users' requests. However, in real-world systems, monitors can fail by collecting information, producing missing or invalid data. This problem can affect the model estimated on the analyzed environment, increasing costs and overloading virtual machines. Therefore, this manuscript presents a new approach that uses data mining methods to replace missing values and configure virtual machines with higher precision.*

**Resumo.** *Mecanismos de virtualização são frequentemente utilizados em ambientes de computação em nuvem para disponibilizar recursos a usuários de maneira escalável, confiável e transparente. A configuração desses mecanismos depende de informações sobre serviços fornecidos. Por exemplo, com o monitoramento de cargas de trabalhos, pode-se calcular a quantidade de máquinas virtuais e o número de processadores necessários para atender todas as requisições. No entanto, em sistemas reais, mecanismos de coleta e monitoramento podem falhar deixando de coletar informações ou produzindo dados inválidos. A ausência ou a imprecisão dessas informações impactam diretamente no modelo estimado, produzindo um ambiente diferente do esperado, aumentando custos e sobrecarregando máquinas virtuais. Assim, este artigo apresenta uma abordagem que utiliza técnicas disponíveis na área de mineração de dados para substituir valores inválidos ou ausentes e configurar ambientes de máquinas virtuais de forma mais eficiente.*

## 1. Introdução

A computação em nuvem fornece uma abstração que permite disponibilizar serviços a clientes possibilitando um melhor aproveitamento dos recursos disponíveis [Xiong and Perros 2009][Zhan et al. 2015]. Visando aumentar a escalabilidade, agilidade e eficiência nessa abstração, provedores de Nuvem adotam mecanismos de

---

\*Trabalho desenvolvido a partir de auxílio financeiro concedido pela UFBA e CNPq nos editais PROPCI-PROPG/UFBA PRODOC/UFBA 2014 (9177) e PIBIC 2015-2016 (8333).

virtualização, os quais permitem atender melhor as demandas impostas pelos clientes [Cherkasova et al. 2007].

De maneira geral, a virtualização oferece meios para compartilhar recursos e serviços, os quais podem ser utilizados para diferentes objetivos [Cherkasova et al. 2007]. Por exemplo, mecanismos de virtualização podem permitir que diferentes sistemas operacionais sejam executados, criando diferentes máquinas virtuais (*Virtual Machine* – VM) sobre um mesmo *hardware*, garantindo total isolamento entre eles, e, ao mesmo tempo, permitindo o compartilhamento de recursos físicos como CPU, memória e dispositivos de entrada e saída.

Em ambientes de larga escala, a forma como máquinas virtuais são alocadas sobre os recursos disponíveis pode influenciar diretamente nos níveis de qualidade de serviço (QoS) oferecidos a usuários [Luo et al. 2010]. Nesse sentido, os parâmetros da alocação são endereçados a partir da (i) quantidade de VMs, (ii) localização (*placement*) e (iii) capacidade em termos de recursos (cpu, memória, entrada-saída).

Por outro lado, alcançar a satisfação dos usuários não é uma tarefa simples, visto que essa satisfação pode depender, por exemplo, de necessidades específicas de cada usuário, da natureza não determinística do ambiente e da escassez dos mecanismos de gerenciamento dos provedores de nuvem. Desse modo, visando fornecer QoS neste tipo de ambiente é necessário que exista um mecanismo capaz de gerenciar de forma adequada a alocação, a criação e configuração de máquinas virtuais na nuvem [Bernstein et al. 2011][Mostajeran et al. 2015].

Com o objetivo de prover QoS em Computação em Nuvem, foi desenvolvida uma arquitetura de metaescalonamento proativa, chamada de **MACC** (**M**etascheduler **A**rchitecture to provide QoS in the **C**loud **C**omputing). O MACC demonstrou ser capaz de atender a maioria desses desafios [Peixoto et al. 2010, Dionisio Filho et al. 2013, Cândido et al. 2013], visto que o mesmo oferece uma plataforma projetada para processar e distribuir os serviços entre diferentes provedores de nuvem com QoS.

Entretanto, observou-se que a qualidade do serviço oferecido pode ser afetada por eventuais erros de transmissão de informação que, se não forem tratados da maneira adequada, podem causar problemas como, por exemplo, na estimação de recursos em ambientes virtuais. Nesse caso, a ausência de informação pode gerar ambientes virtuais subestimados ou com um número excessivo de recursos, caso sejam considerados valores históricos cujo instante de tempo anterior tenha processado um baixo número de instruções ou um número muito superior ao exigido no instante atual.

É importante destacar que o impacto causado pela ausência de dados não é um problema enfrentado exclusivamente por ambiente de computação em nuvem. Segundo [Qu et al. 2009], a ausência de dados é um problema inevitável e pode ser causado por diversas razões como funcionamento inadequado de monitores de informações ou perda de pacote durante a transmissão de dados.

Existem na literatura diferentes formas de trabalhar com a perda de informação a fim de reduzir seu impacto no ambiente analisado. Uma forma comumente utilizada é a replicação de informação, ou seja, a informação é repassada para diferentes recursos garantindo que sempre haverá alguma informação válida disponível. No entanto, além do aumento na transmissão de informação, o problema ainda persistiria se o erro ocor-

resse no monitoramento e coleta da informação. Nessa situação, encontrar mecanismos que permitem estimar um valor substituto para dados ausentes ou inválidos, mantendo as principais características dos valores originais, é um grande desafio.

Dessa forma, este artigo apresenta uma abordagem que busca reduzir o impacto de valores ausentes na estimação de máquinas virtuais em ambientes de computação em nuvem. Visando solucionar esse problema, foi realizada uma Revisão Sistemática da Literatura (*Systematic Literature Review* – SLR) a fim de encontrar métodos de substituição de valores ausentes em conjunto de dados [Kitchenham et al. 2009]. A SLR define critérios para uma rigorosa e bem estruturada busca por artigos publicados em diferentes repositórios, permitindo coletar e avaliar evidências sobre um determinado tema de pesquisa [Kitchenham et al. 2009]. Em resumo, a principal vantagem da SLR é executar uma busca, recuperando não apenas artigos bem conhecidos na área de pesquisa, mas também trabalhos relacionados recentemente publicados.

Neste trabalho, a SLR permitiu encontrar diversas técnicas, comumente utilizadas na área de aprendizado de máquina e processamento de sinais, para substituição de dados ausentes que poderiam ser aplicadas no contexto de alocação de máquinas virtuais em ambiente de computação em nuvem. Dentre os resultados encontrados com a SLR, foram selecionadas as técnicas de substituição de valores ausentes usando interpolação com *Spline* [Ruggiero and da Rocha Lopes 1996, Young and Mohlenkamp 2009] e Análise Espectral Singular (*Singular Spectrum Analysis* – SSA) [Golyandina and Osipov 2007].

Sendo assim, considerando que os métodos selecionados podem substituir valores ausentes que afetam diretamente a qualidade da virtualização na computação em nuvem, foi definida a seguinte hipótese:

*A aplicação de métodos de substituição de dados ausentes permite estimar valores para cargas de trabalho que não foram corretamente coletados do ambiente de computação em nuvem, aumentando, assim, a acurácia na virtualização de recursos*

Os resultados apresentados neste artigo comprovaram a hipótese descrita acima, melhorando a acurácia na estimação de máquinas virtuais, reduzindo custos com a quantidade de máquinas virtuais e o número de processadores necessários para atender as requisições dos usuários.

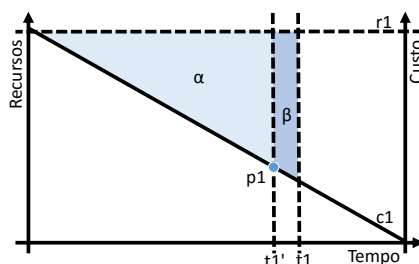
O restante deste artigo está organizado da seguinte forma: na Seção 2, são apresentados os principais conceitos sobre arquitetura de meta-escalonadores utilizados para fornecer qualidade de serviço em ambiente de computação em nuvem; na Seção 3, é apresentada uma breve discussão sobre os métodos selecionados para substituição de valores ausentes; a descrição sobre o ambiente de testes e os conjuntos de dados utilizados nos experimentos são apresentados na Seção 4; por fim, os resultados obtidos e as considerações finais são apresentados e discutidos nas Seções 5 e 6, respectivamente.

## **2. MACC - (*Metascheduler Architecture to provide QoS in Cloud Computing*)**

O MACC é utilizado nesse artigo como uma plataforma de experimentação, capaz de oferecer QoS para um ambiente de computação em nuvem. Este meta-escalonador permite um alto rendimento no total de acessos simultâneos, fornecendo garantias de processamento em tempo real, e de persistência de dados. Esse resultado é alcançado por meio da alocação e do monitoramento de recursos, que operam entre diferentes provedores de

serviço na nuvem, reduzindo custos e gastos de energia, maximizando o compartilhamento de recursos.

Com relação ao modelo de alocação de máquinas virtuais, o MACC lida com o relacionamento entre as variáveis de sistema e o requisito de QoS, de acordo com o gráfico da Figura 1.



**Figura 1. MACC: QoS e Objetivo**

A linha  $r1$  representa o limite físico de recursos, enquanto a linha  $t1$  representa o requisito do SLA referente ao tempo. A linha  $c1$  representa a proporção mínima entre recursos e tempo. As soluções abaixo da linha  $c1$  são impossíveis de reproduzir. A área acima da linha  $c1$  é dita de área factível ( $F$ ). Um resultado dentro da área  $\alpha \cup \beta$  atende os critérios da SLA. Entretanto, os resultados que se encontram em  $F - (\alpha \cup \beta)$  são as soluções com violação da SLA, dado que  $x \in F - (\alpha \cup \beta) \Rightarrow x \in F$  e  $x \notin (\alpha \cup \beta)$ . A reta  $t1'$  representa o fator de segurança para o requisito tempo do SLA. Este fator de segurança evitará que o SLA não seja ferido por elementos externos, como a latência da conexão. Desta forma, a região  $\beta$  deve ser evitada e a área  $\alpha$  é chamada de área segura.

O MACC, por meio dos seus algoritmos de alocação de VM, procura atingir um resultado pertencente à área segura  $\alpha$ . Dentro desta área, qualquer resultado oferecido será satisfatório para o cliente, então o provedor deve alocar as VMs de forma a otimizar o uso dos recursos. Na área  $\alpha$ , o resultado deve convergir para o menor custo e o maior tempo possível. Desta forma, o provedor economiza recursos (maior tempo) e reduz os gastos. O ponto  $p1$  é o resultado ideal, tendo o menor custo e maior tempo dentro da área segura, sendo o objetivo do algoritmo a ser construído.

O MACC tem sido amplamente discutido em trabalhos prévios: i) framework em [Peixoto et al. 2010], ii) abordagem de comunicação [Dionisio Filho et al. 2013], e iii) algoritmos de alocação de máquinas virtuais [Cândido et al. 2013]. Embora o MACC tenha sido analisado em diferentes contextos, em situações onde há presença de dados inválidos (ou ausentes) nas observações de carga de trabalho, a qualidade na alocação de VM na área de computação em nuvem é afetada. É importante ressaltar que não é objetivo deste trabalho avaliar os algoritmos de alocação de máquinas virtuais presentes no MACC nem propor novos métodos de substituição de dados inválidos (ou ausentes), mas apresentar uma nova abordagem para reduzir o impacto desses dados ausentes nos modelos estimados para criação de ambientes virtuais na computação em nuvem.

A próxima seção apresenta os métodos de substituição de dados ausentes que foram utilizados na abordagem proposta neste trabalho.

### 3. Técnicas para Substituição de Valores Ausentes

Nesta seção, são apresentados os resultados obtidos com a execução de uma Revisão Sistemática da Literatura (*Systematic Literature Review* – SLR) [Kitchenham et al. 2009], a qual foi realizada visando encontrar métodos para substituição de dados inválidos que pudessem ser utilizados na abordagem proposta nesse artigo. A execução da SLR foi guiada pela seguinte pergunta principal: “Quais são as técnicas comumente adotadas para resolver o problema de dados ausentes em séries temporais?”. Além disso, definiu-se a string de busca “(time series) AND (gap filling OR missing data)”, a qual foi utilizada no repositório Scopus (<https://www.scopus.com/home.url>). No total, 615 artigos foram recuperados do repositório, sendo que 21 artigos foram selecionados após os critérios de inclusão e exclusão. Os critérios para inclusão de artigos foram: i) o artigo apresenta proposta para tratar dados inválidos em séries temporais; e ii) o artigo é um estudo primário (novos métodos para substituição de dados inválidos); Por fim, os critérios de exclusão adotados foram: i) o artigo apresenta técnicas restritas à problemas específicos; ii) Não é apresentado modelo analítico bem definido; e iii) A avaliação não é satisfatória e não apresenta uma boa revisão da literatura.

A partir da leitura dos artigos retornados pela SLR, foram escolhidas duas técnicas para substituição de valores inválidos. A primeira técnica utiliza o método de interpolação com *spline* cúbica e a segunda utiliza o método de decomposição SSA (*Singular Spectrum Analysis*). A seguir, essas duas técnicas são discutidas em detalhes.

#### 3.1. Spline Cúbica Interpolante

A primeira técnica de substituição de dados inválidos considerada neste trabalho utiliza um método de Spline Cúbica Interpolante e é comumente adotada na área de aprendizado de máquina e processamento de sinais.

De maneira geral, essa técnica analisa um conjunto de observações (ou nós de interpolação)  $X = \{x_0, x_1, \dots, x_n\}$ , os quais foram produzidos por uma regra geradora  $f(\cdot)$  que define o comportamento de um sistema. Uma vez que  $f(\cdot)$  é desconhecida, o objetivo, então, dos métodos de interpolação é estimar uma função aproximada  $f'(\cdot)$  tal que  $f(x_i) = f'(x_i), \forall x_i \in X$  [Ruggiero and da Rocha Lopes 1996].

A interpolação usando spline cúbica obtém uma função estimada  $f'(\cdot)$  combinando diferentes funções polinomiais de grau 3 aplicadas sobre intervalos definidos entre as observações [Young and Mohlenkamp 2009]. Nesse sentido, se a função produz  $n + 1$  pontos, então a função spline cúbica  $S_3(x)$  é definida da seguinte forma:

$$S_3(x) = \begin{cases} C_0(x), & x_0 \leq x \leq x_1 \\ C_i(x), & x_{i-1} \leq x \leq x_i \\ C_n(x), & x_{n-1} \leq x \leq x_n \end{cases} \quad (1)$$

Onde cada  $C_i(\cdot)$  é uma função cúbica na forma geral  $C_i(x) = a_i + b_i x + c_i x^2 + d_i x^3$  e deve passar pelos nós de interpolação:  $C_i(x_i) = y_i$  para  $0 \leq i \leq n - 1$  e  $C_{n-1}(x_n) = y_n$ .

Assim, a continuidade da spline está definida na condição  $C_{i-1}(x_i) = C_i(x_i)$  para  $1 \leq i \leq n - 1$ . Logo, a primeira e a segunda derivada são definidas da seguinte forma:

$$\frac{d}{dx}C_{i-1}(x_i) = \frac{d}{dx}C_i(x_i), \quad \frac{d^2}{dx^2}C_{i-1}(x_i) = \frac{d^2}{dx^2}C_i(x_i)$$

Conseqüentemente, pode-se observar que ao final dos procedimentos obtém-se uma Spline  $S_3(x)$  com a primeira e segunda derivada contínua, ou seja, a curva produzida por  $S_3(x)$  não contém picos nem mudança repentina na curvatura dos nós de interpolação. Essa spline é, então, usada como função  $f'(\cdot)$  e um ponto ausente no instante de tempo  $t$  pode ser estimado, simplesmente, aplicando a função  $S_3(x_t)$ .

### 3.2. Singular Spectrum Analysis – SSA

Uma das limitações do método anterior é a necessidade de estimar parâmetros para os coeficientes do polinômio da spline  $S_3(\cdot)$ . Embora esses coeficientes possam ser estimados usando, por exemplo, o método dos mínimos quadráticos, a suavização gerada por essa spline não considera importantes informações implícitas nos dados como presença de ruído e tendência.

Para solucionar esses problemas, pesquisadores desenvolveram uma técnica não-paramétrica baseado na Análise Espectral Singular (SSA) para substituir valores ausentes. Essa técnica é executada em duas etapas: i) decomposição das observações; e ii) reconstrução substituindo valores inválidos considerando importantes informações implícitas presentes nos componentes decompostos como tendências, sazonalidades, ciclos, periodicidades com amplitudes variadas [Hassani 2007].

Na etapa de decomposição, um conjunto de observações  $X = \{x_1, \dots, x_n\}$ , com  $|X| = n$ , é transformado em uma série com  $L$  dimensões, produzindo uma matriz de trajetórias conforme apresentado na Equação 2, tal que  $K = n - L + 1$ .

$$\mathbf{X} = [X_1 : \dots : X_K] = (x_{ij})_{i,j=1}^{L,K} = \begin{pmatrix} x_1 & x_2 & x_3 & \dots & x_K \\ x_2 & x_3 & x_4 & \dots & x_{K+1} \\ x_3 & x_4 & x_5 & \dots & x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & x_{L+2} & \dots & x_N \end{pmatrix} \quad (2)$$

Em seguida, o produto entre a matriz trajetória e sua transposta,  $\mathbf{S} = \mathbf{X}\mathbf{X}^T$ , é decomposto pelo método SVD (*Singular Value Decomposition*) produzindo seus autovalores e autovalores ortogonais e normalizados [Hassani 2007, Golyandina and Osipov 2007, Golyandina et al. 2001]. O SVD da matriz trajetória pode descrito conforme a Equação 3.

$$\mathbf{X} = \mathbf{E}_1 + \dots + \mathbf{E}_d \quad (3)$$

Na etapa de reconstrução, é realizada um agrupamento das matrizes elementares  $\mathbf{E}_i$  produzidas nos passos anteriores. Para tanto, o SVD apresentado na Equação 3 é particionado em subconjuntos disjuntos:  $X = E_{I_1} + \dots + E_{I_m}$ .

Logo, a saída desse passo é um conjunto de somas de matrizes resultantes da matriz trajetória. O objetivo do agrupamento é diminuir o número de componentes (matrizes elementares) na SVD da matriz trajetória [Golyandina and Osipov 2007]. A decisão de escolha desses conjuntos é fundamental para a aplicação do método SSA e é baseada na propriedade de separabilidade entre conjuntos. A separabilidade é mensurada por correlação ponderada, *i.e.*, seja dois conjuntos de observações  $X_t^{(1)}$  e  $X_t^{(2)}$ , a correlação ponderada entre eles é representada por:

$$\rho_{12}^{(\omega)} = \frac{\langle X_t^{(1)}, X_t^{(2)} \rangle_{\omega}}{\|X_t^{(1)}\|_{\omega} \|X_t^{(2)}\|_{\omega}}$$

A norma da  $i$ -ésima subsérie é dada por  $\|X_t^{(i)}\|_{\omega} = \sqrt{\langle X_t^{(i)}, X_t^{(i)} \rangle_{\omega}}$ , sendo que o produto interno é definido por  $\langle X_t^{(i)}, X_t^{(j)} \rangle_{\omega} = \sum_{c=1}^N \omega_c X_c^{(i)} X_c^{(j)}$  com os pesos  $\omega_c = \min\{c, L, n - c\}$  e  $L \leq n/2$ .

Assim, com base na análise da correlação ponderada pode-se estimar a separabilidade dos componentes [Hassani 2007]. Nesse sentido, [Golyandina and Osipov 2007] criaram uma técnica de substituição de dados inválidos, usando SSA, que extrai componentes aditivos de sinais, como tendência e sazonalidade, e simultaneamente substitui valores inválidos presentes no conjunto de dados. Os experimentos realizados neste artigo utilizam essa técnica para substituição de dados inválidos. De maneira geral, quando valores inválidos estão localizados no final do conjunto de dados (lado direito da série de observações), essa técnica substitui valores inválidos predizendo novos valores com base nos componentes reconstruídos usando SSA. Para os demais valores, na etapa de reconstrução, após escolher um subespaço e uma projeção de vetores no espaço de coordenada de atraso, obtém-se um valor para substituição do primeiro dado inválido a partir da combinação linear de valores anteriores e coeficientes dos componentes principais extraídos com o método SSA. Em seguida, esse passo é repetido até que todos os valores inválidos sejam substituídos [Golyandina and Osipov 2007].

A próxima seção descreve como os experimentos foram organizados, detalhando a criação dos conjuntos de dados e a configuração do ambiente de testes.

#### 4. Organização dos Experimentos

Esta seção apresenta o conjunto de dados e ambiente de simulação que foram utilizados nos experimentos realizados para comprovar a hipótese definida na introdução deste artigo.

O conjunto de dados utilizado neste trabalho é representado por uma série temporal  $x(t) = \{x(1), x(2), \dots, x(T)\}$  com um total de  $T$  observações, *i.e.* a carga de trabalho no instante  $t$  equivale a  $x(t)$  *mips*, tal que  $\forall t \in \{1 \dots T\}$ . Cada observação da série  $x(t)$  foi criada considerando três componentes individuais. O primeiro componente utilizado foi uma função seno, representando um comportamento sazonal com frequência angular igual a  $\pi$ . O segundo componente foi criado utilizando valores aleatórios,  $\epsilon$ , no intervalo  $[0, 1]$  com distribuição uniforme e densidade  $d(t) = \frac{1}{(max-min)}$ . Por fim, um componente de tendência positiva,  $I$ , foi adicionado ao conjunto de dados para simular um crescimento

do número de requisições dos usuários ao longo do tempo. Sendo assim, o conjunto de dados adotado no experimento pode ser representado por  $x(t) = \sin(\pi * t) + \epsilon(t) + I(t)$ .

Após construir a série temporal que contém informações sobre requisições dos usuários em *mips*, definiu-se um número  $W$  que foi utilizado para particionar igualmente os dados em janelas (subconjuntos), representando uma sequência temporal de requisições. Sendo assim, cada janela é formada por um total de  $T/W$  observações. Em seguida, diferentes séries temporais foram produzidas escolhendo janelas aleatoriamente e substituindo os valores de suas observações por um valor inválido (NA).

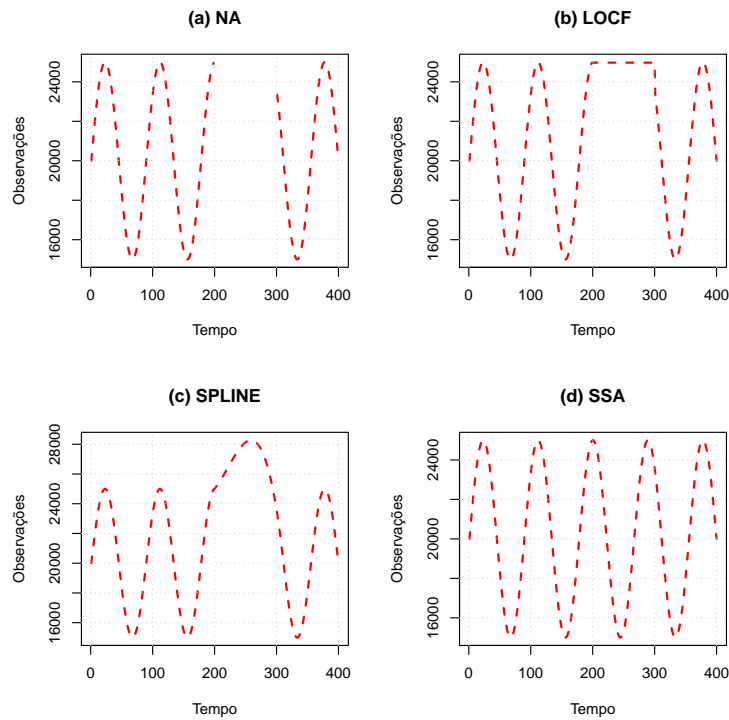
Na etapa seguinte, todas as séries temporais contendo janelas com valores inválidos foram analisadas pelos métodos de interpolação usando *Spline* e *SSA*. Esses métodos analisam as observações válidas de cada série temporal e definem funções aproximadas ( $f'_{spline}(\cdot)$  e  $f'_{ssa}(\cdot)$ , respectivamente) que são utilizadas para estimar novos valores para substituir todos os NAs (valores inválidos). Além da substituição usando esses dois métodos, experimentos foram conduzidos considerando uma função *LOCF*, que substitui valores inválidos pelo último valor válido encontrado, e uma função que, simplesmente, remove NAs, deixando de considerá-los na estimação do modelo usado na virtualização.

É importante destacar, ainda, que optou-se executar os experimentos com dados sintéticos porque sua utilização permite avaliar de maneira mais precisa a abordagem proposta, uma vez que os valores esperados para os dados ausentes são conhecidos *a priori*. Ao inserir NAs na série temporal, os valores válidos foram armazenados para análise futura dos resultados produzidos pelas funções  $f'_{spline}(\cdot)$  e  $f'_{ssa}(\cdot)$ .

Visando simplificar a explicação de todo o processo descrito na preparação do conjunto de testes, considere um *toy problem* que foi criado usando uma série temporal simples, a qual contém apenas um componente senoidal. Esse componente foi criado usando uma função seno com frequência angular igual a  $\sin(\pi t)$ . Essa série temporal contém 400 observações e foi definido um total de  $W = 8$  janelas, ou seja, a série temporal foi dividida em 8 partes contendo 50 observações cada. Neste exemplo, todas as observações das janelas 5 e 6 foram substituídas por valores inválidos (NA), conforme apresentado na Figura 2(a). Em seguida, essa série composta por valores inválidos (NA) nas janelas 5 e 6 foi analisada pelos métodos *LOCF*, *Spline* e *SSA*. As séries obtidas com a substituição de valores ausentes utilizando cada um destes métodos está graficamente representada nas Figuras 2(b), 2(c) e 2(d), respectivamente.

Após substituir valores ausentes utilizando os métodos previamente apresentados, a série temporal é, então, analisada para geração do ambiente de virtualização. Os experimentos realizados com a virtualização apresentados neste artigo foram conduzidos no CloudSim [Calheiros et al. 2011], que é um simulador baseado em eventos e representa o modelo de Computação em Nuvem. Esse simulador foi formulado com base na teoria das filas convencional e possui uma ampla variedade de recursos para cobrir as principais áreas de interesse deste estudo de pesquisa. Para validação da hipótese desse trabalho, considera-se que o simulador utiliza dados coletados sobre o desempenho de aplicações de forma não intrusiva. No contexto deste trabalho, adotou-se os parâmetros apresentados na Tabela 1 para configuração do simulador.

As três principais métricas utilizadas neste trabalho são: i) Tempo de Resposta



**Figura 2. (a) Conjunto de dados criado usando uma senóide com frequência angular igual a  $\pi * t$  e com valores ausentes nas janelas 5 e 6. (b) Os valores ausentes foram substituídos pelo último valor válido (método *LOCF*). (c) Série resultante criada a partir da substituição dos valores ausentes utilizando o método de interpolação com *Spline*. (d) Valores ausentes substituídos por valores obtidos com o método *SSA*.**

**Tabela 1. Variáveis do Experimento**

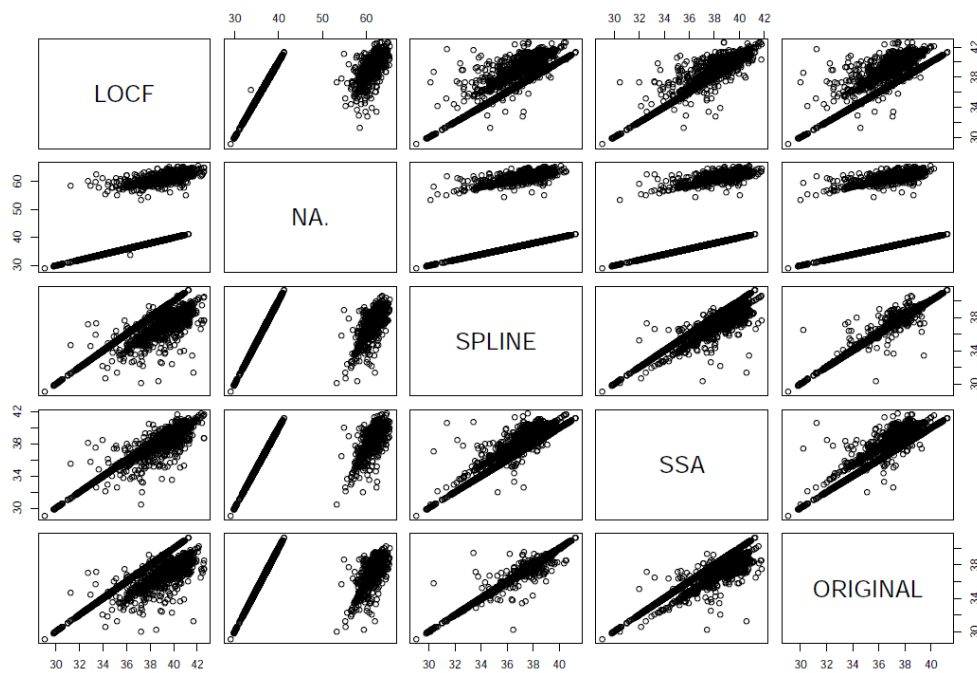
Configuração do Ambiente de Testes	
Usuários	30
Requisições por usuário	400
Total de requisições	12000 requisições
Quantidade de janelas	8
Tamanho de cada janela	1500 requisições
Deadline	40 segundos
Tipo da Carga	Exponencial com rajadas
Valores da Carga	15000 MI — 25000 MI

Médio (TRM) – segundos; ii) Custo Total (C) – \$; e iii) Satisfação (S) – %.

A métrica de **Satisfação** está relacionada ao tempo médio de resposta de um serviço e a frequência de seu atendimento abaixo da deadline imposta. Dessa forma, considera-se que um usuário estará satisfeito com o serviço a ele oferecido quando obtiver uma porcentagem alta de requisições atendidas em média abaixo do limiar de qualidade contratado. Logo, analiticamente, o MACC pode tomar como índice de satisfação do usuário a equação  $S_i = N_i/R_i$ , em que são relacionadas a quantidade total de requisições submetidas pelo i-ésimo usuário  $R_i$  e o número de vezes em que a média do tempo de res-

posta de sistema dessas requisições ficou abaixo do limiar contratado  $N_i$ . Uma requisição é bem atendida quando seu tempo médio de resposta for menor ou igual ao tempo médio de resposta somado a um desvio padrão aceitável contratado. Quanto mais próximo de  $R_i$  for o valor de  $N_i$ , maior será a satisfação proporcionada ao usuário  $i$ . Esta métrica é útil para avaliar o desempenho das técnicas de tratamento de valores ausentes no cumprimento dos níveis de QoS.

De forma geral, as técnicas utilizadas para preenchimento dos valores ausentes apresentam o Tempo de Resposta Médio de acordo com a Figura 3. Esta figura apresenta diagramas de dispersão permitindo a comparação dos resultados obtidos pelas diferentes técnicas e o conjunto de dados esperado (sem valores ausentes). A técnica chamada de ORIGINAL é aquela em que não há dados ausentes, apresentando o comportamento próximo do desejado, ou seja, o conjunto dos tempos de respostas tende ao valor definido do *deadline*. Na última linha da Figura 3 é possível realizar uma leitura comparativa entre as técnicas sugeridas e a ORIGINAL. Dentre as técnicas propostas para utilização, a NA é a que se distânciava mais da solução desejada ORIGINAL. O motivo é ausência total de dados para identificação e criação adequada de máquinas virtuais. Por outro lado, na média, as técnicas SPLINE e SSA são as mais próximas da técnica ORIGINAL.



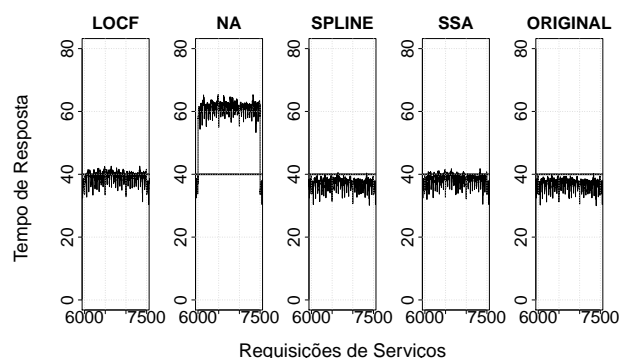
**Figura 3. Técnicas de Tratamento de Valores Ausentes**

Na próxima seção, os resultados e as análises realizadas com a aplicação de todos os conjuntos de dados sintéticos no ambiente de simulação são discutidos em detalhes.

## 5. Resultados

Os resultados apresentados nessa seção foram obtidos considerando a série de observações apresentada na seção anterior, contendo 400 observações divididas em 8 janelas, as quais representam sequências de requisições de usuários em um ambiente de computação em nuvem.

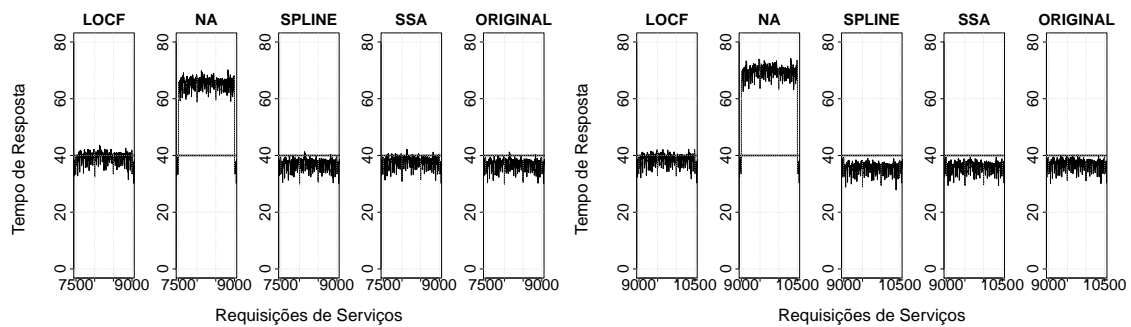
O primeiro resultado foi obtido analisando o meta-escalador sobre um conjunto de observações sendo que todos os valores da janela 5 foram substituídos por valores inválidos (NA). Conforme pode ser visto na Figura 4, não utilizar um algoritmo de substituição de dados inválidos (gráfico NA), não garante que o contrato com o usuário seja cumprido. Além disso, embora o gráfico LOCF apresente melhor resultado do que o gráfico NA, pode-se observar que grande parte das requisições de serviços ficaram acima do deadline estabelecido (40 segundos). Por outro lado, os resultados obtidos com a substituição de dados inválidos usando SPLINE e SSA produziram valores muito próximos do esperado, conforme pode ser visto nos gráficos SPLINE, SSA e ORIGINAL, respectivamente. Lembrando que o gráfico ORIGINAL representa a série de observações antes da inserção de dados inválidos.



**Figura 4. Análise do tempo de resposta e da requisição de serviços usando diferentes técnicas de substituição de dados inválidos para substituição da janela 5.**

De maneira semelhante aos resultados anteriores, nas Figuras 5(a) e 5(b) são apresentados os resultados obtidos analisando o conjunto de dados com a substituição de todos os valores da janela 6 e 7 por dados inválidos. Esse experimento foi importante e serviu para mostrar a estabilidade da abordagem proposta, já que a substituição de valores inválidos na janela 6 e 7 deveria produzir um comportamento semelhante ao observado com a substituição da janela 5.

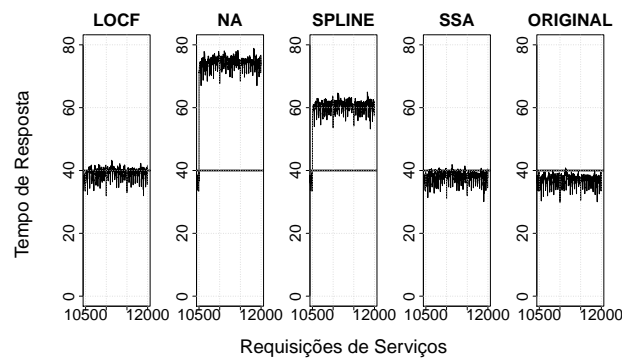
Por fim, foi realizado um experimento no qual todos os dados da última janela foram substituídos por valores inválidos. Esses resultados, apresentados na Figura 6, demonstraram mais uma vez que não realizar um tratamento para substituição de dados inválidos pode levar a violação do deadline para execução de serviços (gráfico NA). No entanto, observou-se que, nesta situação, a técnica de interpolação que usa SPLINE apresentou resultados insatisfatórios semelhante ao não tratamento de dados inválidos, embora o tempo de resposta médio tenha sido o dobro para o gráfico NA. Essa situação ocorreu porque nos outros experimentos realizados, a janela de dados ausentes estava entre duas janelas contínuas de dados válidos, favorecendo a reconstrução do comportamento esperado. Contudo, quando a última janela foi removida, o algoritmo SPLINE não generalizou da maneira adequada, alterando o comportamento original da série. É importante destacar que a técnica SSA permaneceu estável em todos os experimentos, resultando em valores próximos ao esperado (ORIGINAL), garantindo que os tempos de resposta dos serviços não excedessem o deadline estabelecido.



(a) Valores Ausentes na Janela 6

(b) Valores Ausentes na Janela 7

**Figura 5. Análise do tempo de resposta e da requisição de serviços usando diferentes técnicas de substituição de dados inválidos para substituição da janela 6 e 7.**



**Figura 6. Análise do tempo de resposta e da requisição de serviços usando diferentes técnicas de substituição de dados inválidos para substituição da janela 8.**

A Tabela 2 mostra, para cada técnica, os valores monetários gastos com a criação e utilização das máquinas virtuais e também mostra a satisfação dos usuários sobre os serviços oferecidos abaixo da deadline imposta. Todas as técnicas mantêm o mesmo comportamento para o intervalo de janelas  $w = \{1..4\}$ , onde não há valores ausentes. Entretanto, quando são analisadas as janelas  $w = \{5..8\}$ , nota-se que o custo se mantém estável para todas as técnicas, porém é possível verificar que há uma variação substancial em relação a Satisfação dos usuários.

## 6. Considerações Finais

Este trabalho apresentou a utilização de um meta-escalador chamado MACC, responsável pela alocação de máquinas virtuais com garantias de QoS no ambiente de Computação em Nuvem. Nesse ambiente, foi levantada a hipótese que eventuais erros de transmissão de informação poderiam causar problemas na estimação de recursos computacionais a serem utilizados, tais como desperdício de recursos/dinheiro ou até mesmo o não cumprimento dos contratos com os usuários. Dessa forma, foi proposta uma abordagem que utiliza técnicas de substituição de dados inválidos (ou ausentes) para lidar com

**Tabela 2. Tabela de Custo (C) e Satisfação (S)**

Janela	LOCF		NA		SPLINE		SSA		ORIGINAL	
	C(\$)	S(%)	C(\$)	S(%)	C(\$)	S(%)	C(\$)	S(%)	C(\$)	S(%)
1	47,37	99,07	47,37	99,07	47,37	99,07	47,37	99,07	47,37	99,07
2	51,18	99,47	51,18	99,47	51,18	99,47	51,18	99,47	51,18	99,47
3	49,49	99,73	49,49	99,73	49,49	99,73	49,49	99,73	49,49	99,73
4	59,52	99,67	59,52	99,67	59,52	99,67	59,52	99,67	59,52	99,67
5	63,63	60,33	63,58	0	63,63	99,53	63,63	85,93	63,63	99,87
6	67,89	52,47	67,83	0	67,9	99	67,89	94,2	67,9	99
7	71,78	73,6	71,71	0	71,79	100	71,79	100	71,79	99,8
8	76,19	64,27	76,12	0	76,14	0	76,2	92,33	76,2	99,27

esse problema.

Os resultados comprovaram a hipótese deste trabalho, ressaltando a importância da utilização das técnicas de substituição de dados ausentes no ambiente de Computação em Nuvem. A abordagem reduziu o impacto negativo sobre a criação de máquinas virtuais por meio da geração de modelos de carga com a substituição de dados ausentes.

Sobre as técnicas de substituição de dados ausentes, é importante destacar a técnica que utiliza SSA produziu valores mais precisos para dados ausentes e, consequentemente, permitiu estimar modelos de virtualização com maior acurácia. A técnica que utiliza SPLINE não obteve a regularidade para todos os experimentos executados, porém é uma técnica indicada para os casos em que são possíveis a aplicação de um método de interpolação de dados. Já a técnica LOCF é indicada para os cenários onde a construção dos dados de substituição não são viáveis e a utilização do último termo é uma ação com custos de operação reduzida.

Por fim, conclui-se que as técnicas de substituição de dados ausentes foram importantes para estimar valores que não foram corretamente monitorados, aumentando a acurácia na estimação de máquinas virtuais, reduzindo custos com a quantidade de máquinas virtuais e o número de processadores necessários para atender as requisições dos usuários e diminuição do número de contratos não atendidos.

## Referências

- Bernstein, D., Vij, D., and Diamond, S. (2011). An intercloud cloud computing economy technology, governance, and market blueprints. pages 293–299.
- Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A., and Buyya, R. (2011). Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 41(1):23–50.
- Cândido, P., Leite, D., Santana, M., Santana, R., and Peixoto, M. (2013). Alocação proativa de recursos virtualizados aplicada a computação em nuvem. In *SBRC/WCGA - XI Workshop de Computação em Clouds e Aplicações*, SBRC, pages 104–115, Brasília, DF, Brasil.

- Cherkasova, L., Gupta, D., and Vahdat, A. (2007). When virtual is harder than real: Resource allocation challenges in virtual machine based environments. *Hewlett Packard Laboratories, Tech. Rep. HPL-2007-25*.
- Dionisio Filho, M., Peixoto, M. L., Santana, M. J., and Santana, R. H. (2013). P2p routing in the metascheduler architecture to provide qos in cloud computing. *INFOCOMP Journal of Computer Science*, 12(2):1–11.
- Golyandina, N., Nekrutkin, V., and Zhigljavsky, A. A. (2001). *Analysis of time series structure: SSA and related techniques*. CRC press.
- Golyandina, N. and Osipov, E. (2007). The “caterpillar”-ssa method for analysis of time series with missing values. *Journal of Statistical planning and Inference*, 137(8):2642–2653.
- Hassani, H. (2007). Singular spectrum analysis: methodology and comparison. *Journal of Data Science*, 5(2):239–257.
- Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., and Linkman, S. (2009). Systematic literature reviews in software engineering - a systematic literature review. *Information and Software Technology*, 51(1):7–15.
- Luo, M., Zhang, L.-J., and Lei, F. (2010). An insurance model for guaranteeing service assurance, integrity and qos in cloud computing. In *Proceedings of the 2010 IEEE International Conference on Web Services, ICWS '10*, pages 584–591, Washington, DC, USA. IEEE Computer Society.
- Mostajeran, E., Ismail, B., Khalid, M., and Ong, H. (2015). A survey on sla-based brokering for inter-cloud computing. In *Computing Technology and Information Management (ICCTIM), 2015 Second International Conference on*, pages 25–31.
- Peixoto, M. L. M., Santana, M. J., Estrella, J. C., Tavares, T. C., Kuehne, B., and Santana, R. H. C. (2010). A metascheduler architecture to provide qos on the cloud computing. In *ICT '10: 17th International Conference on Telecommunications*, pages 650 – 657, Doha, Qatar. IEEE Computer Society.
- Qu, L., Li, L., Zhang, Y., and Hu, J. (2009). Ppca-based missing data imputation for traffic flow volume: A systematical approach. *Intelligent Transportation Systems, IEEE Transactions on*, 10(3):512–522.
- Ruggiero, M. and da Rocha Lopes, V. (1996). *Cálculo numérico: aspectos teóricos e computacionais*. Makron Books do Brasil.
- Xiong, K. and Perros, H. (2009). Service performance and analysis in cloud computing. In *SERVICES '09: Proceedings of the 2009 Congress on Services - I*, pages 693–700, Washington, DC, USA. IEEE Computer Society.
- Young, T. and Mohlenkamp, M. J. (2009). Introduction to numerical methods and matlab programming for engineers. *Ohio University Department of Mathematics, OH*.
- Zhan, Z.-H., Liu, X.-F., Gong, Y.-J., Zhang, J., Chung, H. S.-H., and Li, Y. (2015). Cloud computing resource scheduling and a survey of its evolutionary approaches. *ACM Comput. Surv.*, 47(4):63:1–63:33.