

Um Método de Seleção de Provedores de Computação em Nuvem Baseado em Indicadores de Desempenho

Lucas Borges de Moraes¹, Adriano Fiorese¹

¹Departamento de Ciência da Computação – (DCC)
Universidade do Estado de Santa Catarina – (UDESC)
Caixa Postal 631 – 89.219-710 – Joinville – SC – Brasil

lucasborges1292@gmail.com, adriano.fiorese@udesc.br

Abstract. *Coping with the successful cloud computing paradigm, the arising of a large number of new cloud computing service providers have been leveraged. This fact hindered the clients' ability to choose among those several cloud computing providers the most appropriate one to attend their requirements and computing needs. This work aims to specify a logical/mathematical method able to select the most appropriate cloud computing providers to the user, based on the analysis of the values of each performance indicator desired by the customer and associated with every cloud computing provider. The method is a three stages algorithm that evaluates, scores, sorts and selects different cloud providers based on the utility of their performance indicators. An example of the method usage is given in order to illustrate its operation.*

Resumo. *O paradigma de computação em nuvem tornou-se bem sucedido e alavancou o surgimento de um grande número de novas empresas provedoras de serviços de computação em nuvem. Esse fato dificultou a capacidade de escolha por parte do cliente de selecionar o provedor de nuvem mais apropriado para suprir as suas necessidades computacionais. Este trabalho visa especificar um método lógico/matemático capaz de selecionar os mais adequados provedores de nuvem para o usuário do método, baseado na análise e pontuação dos valores de seus indicadores de desempenho. O método é estruturado em três etapas e representa um algoritmo que avalia, pontua, ordena e seleciona diferentes provedores, com base na utilidade de seus indicadores. Um exemplo de utilização do método é fornecido de forma a ilustrar seu funcionamento.*

1. Introdução

A modernização da sociedade trouxe consigo a necessidade de dispor recursos computacionais sob demanda, de maneira eficiente, facilitada e com custo acessível. A computação em nuvem tornou-se um paradigma diferenciado de hospedagem e distribuição de serviços computacionais pelo mundo via Internet. Esse paradigma abstrai do usuário a complexa infraestrutura e arquitetura interna provedora do serviço e trouxe o benefício do melhor aproveitamento dos recursos computacionais [Hogan et al. 2013, Zhang et al. 2010].

O sucesso do paradigma da computação em nuvem motivou o surgimento de um grande número de novas empresas como provedoras de computação em nuvem. Essa quantidade crescente de novos provedores tornou a tarefa de escolha e seleção do mais

adequado provedor, para cada necessidade/requisito de seus consumidores, um processo complexo.

A escolha do melhor provedor de nuvem pode ser feita através da avaliação da qualidade geral de um provedor de nuvem. Essa avaliação pode ser feita através da medição sistemática da qualidade de cada um dos indicadores de desempenho ou PIs (*Performance Indicators*) individuais do provedor em questão, resultando em um determinado valor ou *ranking*. Assim, cada provedor possuirá um *ranking* associado calculado em função dos seus PIs. O provedor que apresentar a melhor pontuação através de seus PIs, estará nas primeiras colocações desse *ranking*, e será assim, teoricamente, o mais apropriado para aquele usuário.

Os PIs são ferramentas que permitem uma coleta sintetizada de informações referentes a um determinado aspecto de uma organização e são responsáveis por quantificar (atribuir valor) aos objetos de estudo a serem mensurados. A questão que surge é: Como utilizar tais PIs para mensurar a qualidade de cada provedor para cada usuário? A resposta a essa questão dá-se por meio do método de seleção a ser especificado neste trabalho. Esse método busca utilizar os diferentes tipos de dados (numéricos ou categóricos) coletados de cada provedor de nuvem e armazenados em diferentes PIs, para pontuar uma lista finita de distintos provedores de acordo com os requisitos demandados por cada possível consumidor dos recursos desses provedores. Cada consumidor é um usuário do método, apresentando x requisitos e desejando w diferentes provedores com base nos valores esperados para m PIs de interesse. O método desenvolvido é um algoritmo lógico/matemático capaz de pontuar, ordenar e selecionar os w provedores mais adequados para cada usuário específico. O método proposto é determinístico e agnóstico quanto a quais PIs utilizar, ou seja, o usuário pode solicitar (na expressão de entrada do método) qualquer PI e valor desejado, e havendo provedor compatível, o mesmo será pontuado, ordenado e selecionado.

Este trabalho está organizado da seguinte forma: a seção 2 apresenta trabalhos relacionados à seleção, pontuação e ranking de provedores de nuvem com base em indicadores. A seção 3, apresenta e discute o método proposto que pontua e ordena os provedores de nuvem com base nos PIs desejados pelo usuário. A seção 4 ilustra um exemplo, com dados hipotéticos, de aplicação do método proposto a fim de demonstrar seu funcionamento, apresentando os resultados obtidos. Finalmente, a seção 5 apresenta as considerações finais do trabalho.

2. Trabalhos Relacionados

Diferentemente dos trabalhos já desenvolvidos, esse trabalho apresenta uma solução determinística para o problema da seleção de provedores de nuvem, que leva em conta a análise de indicadores tanto qualitativos quanto quantitativos, considerando diferentes níveis de importância atribuídos pelo próprio usuário de acordo com suas necessidades computacionais. Essa análise também leva em consideração uma abordagem matemática que leva em consideração não só a priorização de certos indicadores, como também tolerâncias máximas aceitáveis aos valores dos indicadores desejados.

Por outro lado, em [Sundareswaran et al. 2012], é proposta uma nova arquitetura de corretagem na nuvem, onde os corretores são responsáveis pela seleção do serviço apropriado para cada usuário/cliente (consumidor). O corretor tem um contrato com os provedores, recolhendo suas propriedades (os indicadores de desempenho), e com os con-

sumidores, recolhendo suas exigências de serviço. Ele analisa e indexa os prestadores de serviço de acordo com a semelhança de suas propriedades. Ao receber uma solicitação de seleção de serviço o corretor procurará o índice para identificar uma lista ordenada dos provedores candidatos com base em quão bem eles correspondem às necessidades dos utilizadores.

Um *framework* chamado “SmiCloud” é apresentado em [Garg et al. 2013]. Ele é responsável por mensurar a qualidade de serviço (QoS) de provedores de nuvem e ranqueá-los com base nessa qualidade calculada. A qualidade está diretamente relacionada aos valores de cada métrica do *Service Measure Index*(SMI) [CSMIC 2014] classificados em funcionais e não funcionais. O trabalho utiliza o método *Analytical Hierarchical Process* (AHP) [Saaty 2004] para ponderação no cálculo da qualidade e ranqueamento dos provedores.

O *framework* desenvolvido em [Baranwal and Vidyarthi 2014] apresenta uma expectativa quanto as métricas de QoS (baseadas também no SMI) que o provedor deve ter e as repassa a um corretor que o auxilia na escolha do mais adequado. Ele utiliza um método de votação para o ranqueamento (idêntico a [Shirur and Swamy 2015]) em que são levadas em consideração as exigências do usuário. Cada métrica (que pode assumir diferentes tipos - numérico, booleano, um intervalo, um conjunto não ordenado) irá funcionar como um eleitor e os provedores de nuvem serão os candidatos para elas.

O trabalho desenvolvido em [Achar and Thilagam 2014] apresenta uma abordagem baseada em um corretor de serviços para selecionar o mais adequado provedor de nuvem com base na medição da qualidade de serviço prestada de cada provedor, priorizando aqueles mais adequados às necessidades de cada solicitação ao corretor.

Em [Shirur and Swamy 2015], um *framework* é especificado para quantificar, através de métricas de Qualidade de Serviço (QoS), a eficiência de diferentes provedores de computação em nuvem, ranqueando-os. O *framework* divide as métricas de QoS em duas categorias: as dependentes da aplicação (confiabilidade, disponibilidade, segurança, *data centers*, custo, sistemas operacionais suportados, plataformas suportadas, tempo de resposta, *throughput* e eficiência) e as dependentes do usuário (reputação, interface do cliente, teste gratuito, certificação, sustentabilidade, escalabilidade, elasticidade e experiência do usuário).

O trabalho desenvolvido em [Wagle et al. 2015] propõe um modelo de avaliação que verifica a qualidade e o status dos serviços em nuvem fornecidos por provedores de nuvem. O dados são obtidos por auditores de nuvem e são visualizados em um mapa de calor ordenado pelo desempenho de cada provedor, mostrando-os em ordem decrescente de qualidade de serviço global prestado.

3. O Método de Seleção Proposto

O método proposto visa auxiliar a tomada de decisão de um usuário que deseja selecionar o mais adequado provedor de computação em nuvem por meio de valores de PIs de interesse e a importância de cada PI para o usuário em questão. Para tal, o usuário ou um terceiro (ex: o servidor onde está hospedado o método de seleção) deve possuir inicialmente uma base de dados contendo uma lista com distintos provedores de nuvem, cada qual com seu respectivo conjunto de PIs e seus valores. A Figura 1 apresenta uma visão

geral do método de seleção de provedores a ser descrito nesta seção. A entrada de dados (*Inputs*) corresponde a uma lista $P = [P_1, P_2, P_3, \dots, P_n]$ com n distintos provedores de nuvem candidatos a seleção, cada qual com, inicialmente, M diferentes PIs associados (cujos valores são conhecidos e estão armazenados no banco de dados do método) e uma expressão de entrada gerada pelo usuário, contendo os m PIs de interesse (subconjunto dos PIs cadastrados no banco de dados do método) e o nível de importância de cada um (L níveis possíveis, arbitrado pelo usuário).

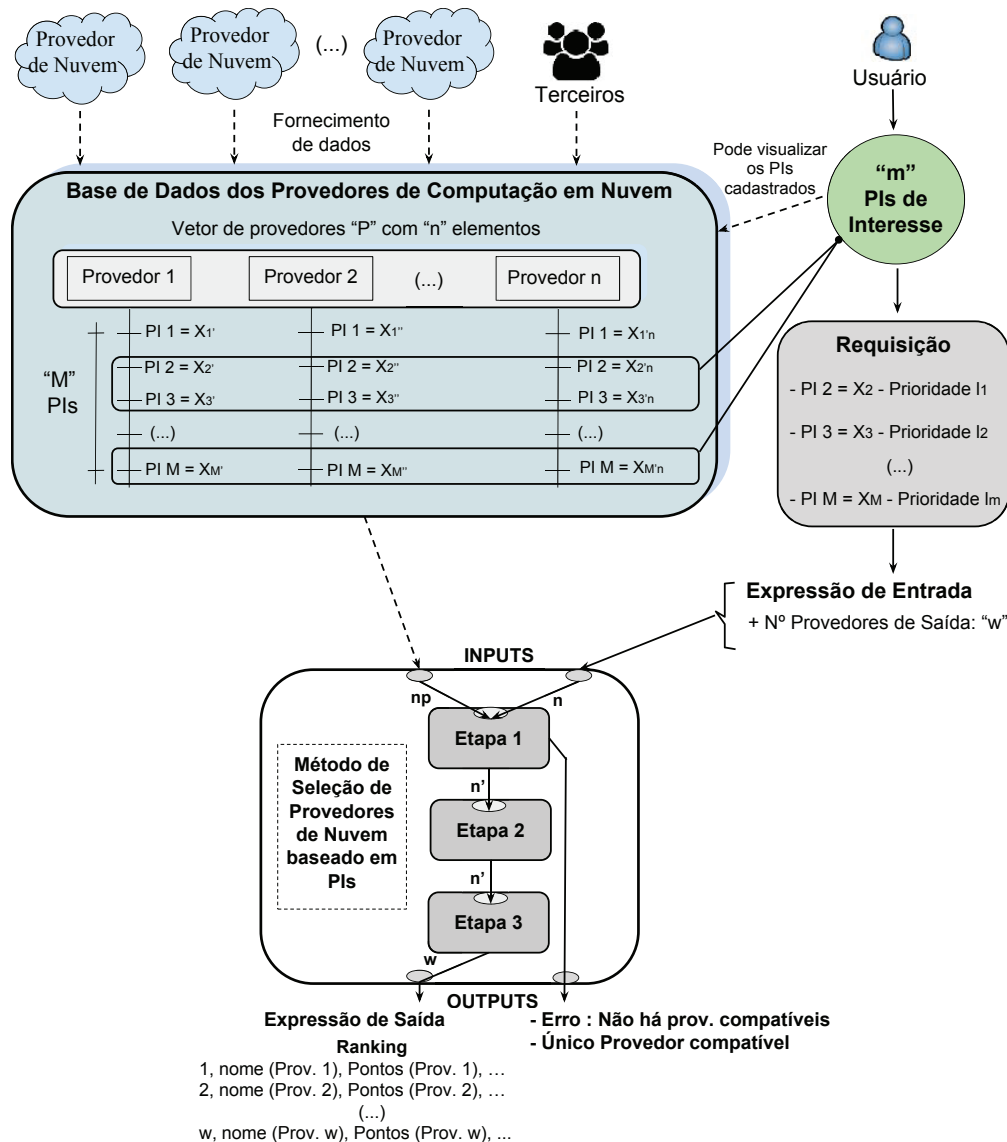


Figura 1. Método de seleção de provedores de nuvem baseado em PIs

Essa lista inicial P será diminuída, logo na primeira etapa do método, com base na expressão de entrada do usuário (que deve apresentar o identificador de cada PIs de interesse, assim como seus respectivos valores desejados e sua respectiva importância e eventual tolerância) e terá n' elementos, com $n \geq n'$. Se $n' = 0$, nenhum provedor disponível no banco de dados é compatível com o usuário, e assim interrompe-se o processo com uma mensagem de erro; se $n' = 1$ há um único provedor compatível, que é retor-

nado; se $n' > 1$ segue-se com as etapas seguintes. A saída de dados esperada (*Outputs*), exceto as condições especiais mencionadas, será uma lista com os w provedores mais bem pontuados pelo método.

O método não restringe quais PIs o usuário deve utilizar para selecionar seu provedor. Contudo, para a computação em nuvem, é possível utilizar um conjunto de PIs específicos, como aqueles presentes no *framework Service Measure Index* (SMI) [CSMIC 2014], e outros, presentes em trabalhos como: [Sundareswaran et al. 2012, Garg et al. 2013, Baranwal and Vidyarthi 2014, Achar and Thilagam 2014, Wagle et al. 2015, Shirur and Swamy 2015].

O método proposto é dividido em três etapas principais:

1. Eliminação de provedores incompatíveis ao usuário;
2. Pontuação dos PIs quantitativos e qualitativos para cada nível de importância; e
 - (a) Pontuação dos PIs quantitativos.
 - (b) Pontuação dos PIs qualitativos.
3. Cálculo da pontuação final geral e ranqueamento dos provedores.

Cada uma dessas etapas será discutida nas subseções a seguir.

3.1. Etapa 1 – Eliminação dos Provedores Incompatíveis

A Figura 2 ilustra o que ocorre nesta primeira etapa do método de seleção. A Etapa 1 remove da lista inicial de provedores candidatos (lista P), todo provedor incompatível ao usuário (conceito e ser definido nessa seção), gerando assim, uma nova lista P' com n' diferentes provedores. As pontuações de cada provedor ocorrem efetivamente na segunda e terceira etapa.

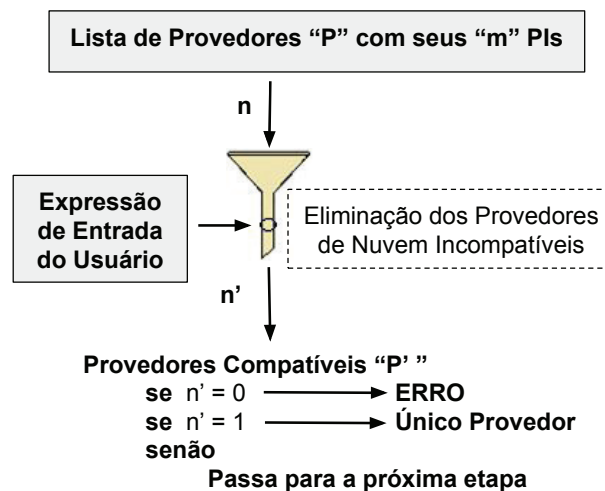


Figura 2. Etapa 1 – Eliminação de provedores incompatíveis

Dependendo do valor de n' o processo de pontuação e seleção prossegue ou finaliza. Se há mais de um provedor compatível para a seleção ($n' > 1$), inicia-se a próxima etapa, caso contrário, ocorrem as saídas especiais identificadas na Figura 2.

O conceito de compatibilidade ou incompatibilidade de um provedor de nuvem está associado com o atendimento ou não dos requisitos do usuário. Esses requisitos são

modelados como valores desejados dos PIs de interesse do usuário. Nesse sentido, para o completo entendimento do que significa esse atendimento, é necessário uma compreensão mais detalhada de como são considerados e tratados os PIs no método proposto.

Um PI pode ser classificado como essencial ou não essencial pelo usuário. Um PI essencial possui importância máxima, indicando que se ele não for atendido com aquele valor informado, o usuário não conseguirá atingir seus objetivos. Logo, será considerado incompatível todo provedor que não atender a todos os PIs essenciais do usuário.

Os PIs não essenciais possuem diferentes níveis de importância, que podem variar entre os níveis “Alto” e “Baixo”. Quando um provedor de nuvem não atende um PI não essencial, isso não incapacita o usuário de atingir seus objetivos, mas pode prejudicá-lo (diretamente relacionado a importância estipulada pelo usuário). Por exemplo, não atender um PI de nível “Alto” significa que usuário ficará seriamente prejudicado em atingir seus objetivos; por outro lado, não atender um de nível “Baixo”, implica em que o impacto negativo será mais tolerável (ex: PIs opcionais).

Os PIs podem ser classificados em duas classes [Jain 1991]: quantitativo (discreto ou contínuo) e qualitativo (ordenado ou não ordenado). Se o PI for quantitativo, há três classificações quanto ao comportamento [Jain 1991] da expectativa que seus valores apresentam para as necessidades dos usuários: HB ou *Higher is Better* (maior é melhor), indicando que o provedor de nuvem pode apresentar valores maiores que desejado pelo usuário; LB ou *Lower is Better* (menor é melhor), indicando que o provedor pode apresentar valores menores que o desejado pelo usuário e NB ou *Nominal is Best* (nominal é melhor) implicando que valores devem ser idênticos ao ideal. Assim, seja PI_{ij} o j-ésimo PI quantitativo de interesse de um usuário e pertencente ao i-ésimo provedor de P (P_i). Se PI_{ij} armazena o valor x e PI_{ij} atende ao valor y , especificado pelo usuário, então conclui-se que:

$$\begin{cases} x \geq (y - t_j) & \text{se } PI_{ij} \in HB \\ x \leq (y + t_j) & \text{se } PI_{ij} \in LB \\ x = (y \pm t_j) & \text{se } PI_{ij} \in NB \end{cases} \quad (1)$$

Em que t_j representa a tolerância (a x) por parte do usuário para o j-ésimo PI quantitativo de interesse, quando o valor desejado pelo mesmo é y . A tolerância padrão é zero, para cada PI, mas ela pode ser ajustada pelo usuário via expressão de entrada, para cada PI desejado.

Se PI_{ij} for qualitativo, ele pode ser ordenado ou não ordenado [Jain 1991]. Essa subclassificação advém da existência ou não de uma relação de ordem entre os possíveis valores assumíveis de um PI qualitativo. Assim, caso o PI seja qualitativo não ordenado (não existe relação de ordem), a regra é simples: se o valor de $PI_{ij}(x)$ é aquele que o usuário deseja, então PI_{ij} atende y , caso contrário não atende. Caso o PI seja ordenado, cada valor (categoria ou classe) de PI_{ij} possui uma certa relação com os outros, graduando de um nível mais baixo até um nível mais alto. Se o usuário especificar um valor de nível mais baixo, um valor de nível mais alto pode também satisfazê-lo, depende do PI em questão.

Para resolver tal problema será criada uma nomenclatura que indicará se um PI qualitativo ordenado possui tolerâncias para categorias abaixo (*Lower*) e/ou acima

(*Higher*) da categoria desejada, semelhante a nomenclatura de [Jain 1991]: HT ou *Higher is Tolerable* (Categorias acima da desejada são toleráveis), LT ou *Lower is Tolerable* (Categorias abaixo da desejada são toleráveis) e HLT ou *Higher and Lower is Tolerable* (Categorias acima e abaixo da desejada são toleráveis). Todas essas tolerâncias podem ser especificadas pelo usuário do método via expressão de entrada. Caso nada seja informado, o padrão usado é NB (sem tolerância alguma). Assim, dado um PI qualitativo PI_{ij} , que armazena o valor x (*string*-categoria), e PI_{ij} é o j -ésimo PI qualitativo de interesse pertencente ao i -ésimo provedor de P (P_i). Se PI_{ij} atende ao valor y , especificado pelo usuário, conclui-se que:

$$\left\{ \begin{array}{l} x = y \text{ se } PI_{ij} \text{ é não ordenado OU } PI_{ij} \in NB \\ x \geq y \text{ se } PI_{ij} \text{ é ordenado E } PI_{ij} \in HT \\ x \leq y \text{ se } PI_{ij} \text{ é ordenado E } PI_{ij} \in LT \\ PI_{ij} \text{ é ordenado E } PI_{ij} \in HLT \end{array} \right. \quad (2)$$

Logo, levando em consideração a premissa de atendimento e o conceito de incompatibilidade, ao fim dessa etapa inicial restará uma lista com somente provedores de nuvem compatíveis com as exigências de PIs essenciais atendidas. Na próxima etapa é necessário pontuar esses provedores de acordo com os demais PIs desejados pelo usuário.

3.2. Etapa 2 – Pontuação dos PIs para cada Nível de Importância

A segunda etapa visa pontuar os níveis de importância para cada provedor individualmente, de acordo com a utilidade de cada um dos seus PIs, sejam eles quantitativos ou qualitativos. A Equação 3 apresenta a pontuação por nível de importância, como a razão entre a pontuação dos PIs quantitativos e qualitativos que pertencem àquele nível l , e o total de PIs pertencentes ao nível l (quantidade m_l).

$$P_l(P_i) = \frac{\sum_{k=1}^m (P_{qt}(PI_k) + P_{qt}(PI_k))}{m_l} \quad [PI_k \in l] \quad (3)$$

A Etapa 2 finaliza quando são pontuados todos os L níveis de importância, para cada um dos n' provedores disponíveis. Independentemente do nível, a pontuação individual para um PI quantitativo e um qualitativo é feita de formas diferentes.

3.2.1. Pontuação dos PIs Quantitativos

A pontuação do j -ésimo PI quantitativo de um provedor i (PI_{ij}), será 0, caso não atenda ao valor numérico y , especificado pelo usuário. Caso o valor atenda, ele será pontuado proporcionalmente ao quão útil esse valor é em relação a todos os outros provedores disponíveis na lista de seleção. A função de avaliação de um PI quantitativo está representada na Equação 4. Ela retorna sempre um valor real normalizado entre 0 e 1.

$$P_{qt}(PI_j) = \begin{cases} 0, \text{ se } valor(PI_{ij}) \text{ não atende } y \\ C_1 + C_2 * \frac{valor(PI_{ij})-y}{X_{max}-y}, \text{ se } PI_{ij} \in HB \\ C_1 + C_2 * \frac{y-valor(PI_{ij})}{y-X_{min}}, \text{ se } PI_{ij} \in LB \\ C_1 + C_2 * \frac{t_j-|y-valor(PI_{ij})|}{t_j}, \text{ se } PI_{ij} \in NB \end{cases} \quad (4)$$

O número X_{max} é o maior valor encontrado dentre todos os n' provedores na lista P' para aquele PI_{ij} ; assim como X_{min} é o menor valor e t_j a distância máxima tolerada do ponto ótimo (y) para um PI do tipo NB (desde que o mesmo atenda y , ou seja, pertença ao intervalo de aceitação $[y - t_j; y + t_j]$). As constantes reais (parâmetros) C_1 e C_2 pertencem ao intervalo aberto normalizado entre $]0, 1[$, assim, $C_1 + C_2 = 1$, obrigatoriamente. O valor de t_j é configurado pelo usuário para cada PI de interesse.

A constante real C_1 pondera a pontuação dada a correspondência (*matching*) mínima desejada (incluindo as tolerâncias associadas ao PI, se existirem) entre o valor do provedor (x) e o valor desejado (y), baseando-se no tipo de PI (HB, LB ou NB) em análise. A constante real C_2 pondera a pontuação dada ao quanto esse valor de PI se sobressai ao mínimo desejado, ou seja, o valor x é, na prática, melhor que o valor y desejado.

3.2.2. Pontuação dos PIs Qualitativos

Para os PIs qualitativos ao invés de valores numéricos tem-se valores categóricos (*string*). Em relação aos PIs qualitativos não ordenados: se a categoria é a que o usuário especificou, então o PI recebe pontuação máxima 1; caso contrário recebe 0. Para os PIs qualitativos ordenados, a pontuação depende da tolerância suportada e informada pelo usuário (HT, LT ou HLT) em relação ao valor (categoria) ofertado pelo provedor. Categorias de níveis superiores e/ou inferiores à categoria desejável y podem ser toleráveis ao usuário, podendo assim pontuar.

A Figura 3.2.2 ilustra como a pontuação é influenciada pelo tipo de tolerância associada a um PI qualitativo ordenado (HT, LT e HLT), como se especificam os níveis das categorias (valores) e como se calcula a distância entre esses níveis, para nove distintas categorias hipotéticas. O nível da categoria y e das categorias toleradas estão sublinhadas na Figura 3.2.2. O nível de uma categoria é representado por um número inteiro positivo, que inicia em 1 e finaliza com o total de categorias disponíveis para aquele PI, ordenados de forma crescente (níveis inferiores - números menores; níveis superiores - números maiores). A distância entre as categorias A (valor do PI qualitativo do provedor) e y (valor desejado) é o módulo da diferença entre o nível de ambas, tendo papel relevante na pontuação de um PI qualitativo ordenado. A pontuação de uma categoria tolerada (pontuação não nula) é uma constante (nomeada C_3 , valor real menor que 1) multiplicada pela distância normalizada entre os níveis das categorias A e y . Assim, a categoria desejada recebe pontuação máxima 1, as categorias vizinhas (superior e inferior) terão pontuação igual a C_3 . As demais variam no intervalo aberto de $]0, C_3[$ conforme a Equação 5. Com K_1 e K_2 sendo, respectivamente, o número total de categorias acima e abaixo do nível da categoria y .

$$P_{ql}(PI_{ij}) = \begin{cases} 1, & \text{se } valor(PI_{ij}) = y \\ C_3 * \frac{K_1 - dist(valor(PI_{ij}), y) + 1}{K_1}, & \text{se } PI_{ij} \in HT \\ C_3 * \frac{K_2 - dist(valor(PI_{ij}), y) + 1}{K_2}, & \text{se } PI_{ij} \in LT \\ C_3 * \frac{K_1 + K_2 - dist(valor(PI_{ij}), y) + 1}{(K_1 + K_2)}, & \text{se } PI_{ij} \in HLT \\ 0, & \text{Caso contrário} \end{cases} \quad (5)$$

A = Valor oferecido = **Baixo** = nível 3 **K1** = Categorias acima Alto = 2
 y = Valor desejado = **Alto** = nível 7 **K2** = Categorias abaixo Alto = 6

Pontuação: **HT:** 0 **LT, HLT:** $C_3 * \text{norm}(\text{Dist}(A,y))$ **NB:** 0

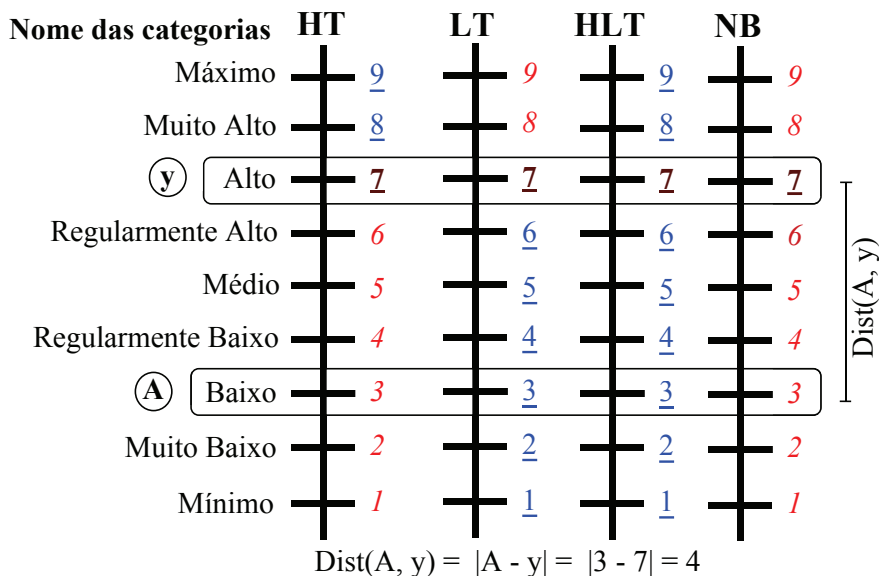


Figura 3. Relação entre pontuação e tipo de tolerância associada a um PI qualitativo ordenado hipotético com 9 categorias

A constante real C_3 representa a pontuação máxima que um categoria tolerável (diferente da desejável y , mas que se enquadra nas tolerâncias associadas para aquele PI) pode assumir. Logo, quanto menor o valor de C_3 mais agressiva é a penalidade (perda de pontuação) aplicada a todo e qualquer PI_{ij} , cuja categoria x diverge do ponto ótimo desejado y .

3.3. Etapa 3 – Cálculo da Pontuação Final de cada Provedor

Por meio das duas etapas anteriores, foram calculadas as pontuações para cada um dos L diferentes níveis de importância, para cada provedor da lista P' . Assim, a pontuação final do provedor de nuvem será a média aritmética ponderada das pontuações dos diferentes níveis de importância, cujos pesos são diretamente proporcionais ao nível de importância que aquela pontuação se refere. A Equação 6 apresenta a pontuação final de um provedor P_i .

$$P_{final}(P_i) = \sum_{l=1}^L (\alpha_l * P_l(P_i)) \tag{6}$$

É importante notar que obrigatoriamente $\alpha_1 + \dots + \alpha_L = 1$, normalizando a pontuação de cada provedor entre 0 e 1. Uma técnica para calcular cada α_l é utilizar uma matriz de julgamentos [Saaty 2004]. Nessa matriz, para esse trabalho, os elementos a serem relacionados são os níveis de importância para os PIs.

A Tabela 1 apresenta uma possível matriz de julgamentos para 4 níveis de importância ($L = 4$). Os valores atribuídos são baseados na escala de Saaty [Saaty 2004]. A

matriz de julgamento representa a análise pareada dos elementos que a compõe, e nesse caso, indica quão importante é o elemento da linha i em relação a coluna j . Assim, com essa metodologia obtém-se a diagonal igual a 1 e as inversões observáveis. A Tabela 1 representa a matriz de julgamento original idealizada para esse trabalho. Na última linha é apresentada a soma das colunas, antecipando o próximo passo que é a normalização dessa matriz através da divisão dos valores da coluna pela soma dos valores da coluna. Com a normalização, o cálculo dos pesos para cada nível dá-se pela média aritmética dos valores das linhas, da tabela normalizada. A Tabela 2 representa a Tabela 1 normalizada e já com os pesos calculados para cada nível de importância.

Tabela 1. Matriz de julgamento: Relações de importância para 4 diferentes níveis de importância

| Nível de importância | Essencial | Alto | Médio | Baixo |
|----------------------|---------------|---------------|---------------|-------|
| Essencial | 1 | 2 | 4 | 9 |
| Alto | $\frac{1}{2}$ | 1 | 2 | 6 |
| Médio | $\frac{1}{4}$ | $\frac{1}{2}$ | 1 | 3 |
| Baixo | $\frac{1}{9}$ | $\frac{1}{6}$ | $\frac{1}{3}$ | 1 |
| Soma Coluna | 1,8611 | 3,6667 | 7,3333 | 19,00 |

Tabela 2. Matriz de julgamento normalizada para 4 níveis de importância

| Nível de importância | Essencial | Alto | Médio | Baixo | Pesos |
|----------------------|-----------|--------|--------|--------|---------------|
| Essencial | 0,5373 | 0,5455 | 0,5455 | 0,4737 | 0,5255 |
| Alto | 0,2687 | 0,2727 | 0,2727 | 0,3158 | 0,2825 |
| Médio | 0,1343 | 0,1364 | 0,1364 | 0,1579 | 0,1413 |
| Baixo | 0,0597 | 0,0455 | 0,0455 | 0,0526 | 0,0508 |

Assim, após as verificações de consistência da matriz de julgamentos [Saaty 2004], temos os pesos para cada nível de importância ($l \in L$), de acordo com a última coluna da Tabela 2. A Equação 7 apresenta as incógnitas α_j da Equação 6, resolvidas por meio da Tabela 2.

$$P_{final}(P_i) = 0,5255 * P_{Essencial}(i) + 0,2825 * P_{Alto}(i) + 0,1413 * P_{Medio}(i) + 0,0508 * P_{Baixo}(i) \quad (7)$$

Após a pontuação de todos os provedores, a lista de provedores compatíveis é ordenada em ordem decrescente de pontuação. Com isso, os w primeiros provedores dessa lista são retornados ao usuário conforme solicitado.

4. Utilizando o Método Proposto

Após especificado o método, é necessário expor alguns exemplos quanto a aplicação prática do mesmo sobre um conjunto de dados reais ou hipotéticos, para mostrar o seu funcionamento e sanar dúvidas quanto aos procedimentos e resultados obtidos. Essa seção visa aplicar o método especificado sobre um possível conjunto de dados. A Tabela 3 é um exemplo de dados representativos de provedores de nuvem candidatos que podem ser utilizados no método de seleção descrito. Ela contém seis provedores fictícios disponíveis, cada qual com sete PIs.

Tabela 3. Exemplo de dados de PIs para seis provedores de nuvem

| PI/Provedor | P_1 | P_2 | P_3 | P_4 | P_5 | P_6 |
|--|-------|-------|-------|-------|---------|-------|
| RAM (Gb) | 4 | 2 | 4 | 8 | 16 | 4 |
| Armz (Gb) | 10,00 | 50,00 | 80,00 | 15,00 | 5,00 | 3,00 |
| Custo (R\$/h) | 0,30 | 0,50 | 1,00 | 1,50 | 1,20 | 1,00 |
| Disp (%) | 99,90 | 95,00 | 88,00 | 99,00 | 90,00 | 90,00 |
| NI | 5 | 3 | 9 | 12 | 10 | 8 |
| Seg | Médio | Médio | Alto | Baixo | Médio | Médio |
| SOS | L, W | L, M | W | L | L, M, W | W |
| Legenda | | | | | | |
| RAM: Quantidade média de memória RAM disponibilizada em Gigabytes. | | | | | | |
| Armz: Quantidade média de memória para armazenamento de dados em Gigabytes. | | | | | | |
| Custo: Custo médio do serviço em reais por hora. | | | | | | |
| Disp: Disponibilidade média do serviço por ano em percentagem. | | | | | | |
| NI: Número total de tipos de instâncias (VMs) disponibilizadas. Inteiro > 1 . | | | | | | |
| Seg: Nível estimado de segurança e privacidade da informação. Possui 3 categorias: Alto, Médio e Baixo. | | | | | | |
| SOS: Sistemas operacionais suportados. Possui 3 categorias: Windows (W), Linux (L) e Mac (M). | | | | | | |

O primeiro passo é identificar a natureza de cada PI e verificar quais atendem e quais não atendem os valores desejados pelo usuário. Assim, inicialmente é necessário que de alguma maneira (ex: um banco de dados mantido por especialistas ou pelo usuário do método) o método proposto reconheça que o PI “NI” seja quantitativo discreto com função de utilidade HB; “Custo”, quantitativo contínuo LB; “RAM”, quantitativo discreto HB; “Armz”, quantitativo contínuo HB; “Disp”, quantitativo contínuo HB; “Seg”, qualitativo ordenado NB; e “SOS”, qualitativo não ordenado. Se o usuário buscar um provedor de nuvem que atenda os requisitos conforme a Tabela 4, temos a Tabela 5. Ela mostra os valores desejáveis (de acordo com as funções de utilidade associadas) e toleráveis (de acordo com os valores de tolerância associados às funções de utilidade fornecidas pelo usuário) para cada PI, bem como outros valores úteis necessários para a pontuação final de cada provedor.

Tabela 4. Exemplo de expressão de entrada do usuário

| PI e valor desejado | Tolerância | Nível de importância |
|---------------------|------------|----------------------|
| $RAM = 4$ | – | Essencial |
| $Armz = 5$ | 0,50 | Essencial |
| $Custo = 1,00$ | 0,10 | Alto |
| $Disp = 90$ | 0,50 | Alto |
| $NI = 8$ | 1 | Médio |
| $Seg = Medio$ | HT | Médio |
| $SOS = W$ | – | Baixo |

Por meio dessa análise, já é possível determinar se há provedores incompatíveis (Etapa 1). É incompatível qualquer provedor que não atenda todos os PIs essenciais (“RAM” e “Armz”). Um PI atende certo valor desejado, se o seu valor está no intervalo de

Tabela 5. Análise dos PIs apresentados no exemplo

| PI | Valores desejáveis | Valores toleráveis | Valores Mín / Max |
|-------|--------------------|--------------------|---------------------------------|
| RAM | $[4, +\infty)$ | – | $X_{max} = 16 \text{ GB}$ |
| Armz | $[5, 0; +\infty)$ | $[4, 5; 5, 0)$ | $X_{max} = 80, 0 \text{ GB}$ |
| Custo | $[0; 1, 00]$ | $(1, 00; 1, 10]$ | $X_{min} = 0, 30 \text{ R\$/h}$ |
| Disp | $[90, 0; 100, 0]$ | $[89, 5; 90, 0)$ | $X_{max} = 99, 9\%$ |
| NI | $[8, +\infty)$ | 7 | $X_{max} = 12$ |
| Seg | Médio | Alto | – |
| SOs | W | – | – |

valores desejáveis ou, pelo menos, no intervalo de valores toleráveis, ambos identificados na Tabela 5. Com base nisso, foi construída a Tabela 6, onde (●) informa que o PI atende ao valor desejado pelo usuário e (○) não atende.

Tabela 6. Identificação se o PI de um provedor atende o valor desejado pelo usuário

| PI / Provedor | P_1 | P_2 | P_3 | P_4 | P_5 | P_6 |
|---------------|-------|-------|-------|-------|-------|-------|
| RAM | ● | ○ | ● | ● | ● | ● |
| Armz | ● | ● | ● | ● | ● | ○ |
| Custo | ● | ● | ● | ○ | ○ | ● |
| Disp | ● | ● | ○ | ● | ● | ● |
| NI | ○ | ○ | ● | ● | ● | ● |
| Seg | ● | ● | ● | ○ | ● | ● |
| SOs | ● | ○ | ● | ○ | ● | ● |

Por meio da Tabela 6 conclui-se que o Provedor 2 não atende ao PI essencial “RAM” de 4 GB ou mais e o Provedor 6 não atende ao PI essencial “Armz” de 4,5 GB ou mais. Assim, os provedores 2 e 6 devem ser removidos da lista de provedores aptos/compatíveis para a seleção. A seguir (Etapa 2), deve-se pontuar os PIs quantitativos (Equação 4) e qualitativos (Equação 5) para P_1, P_3, P_4 e P_5 . A pontuação deve ser feita por nível de importância de acordo com a Equação 3. A Tabela 7 apresenta os resultados finais calculados para os quatro níveis de importância mostrados. As constantes utilizadas foram: $C_1 = 0.7$, $C_2 = 0.3$ e $C_3 = 0.7$.

Tabela 7. Pontuação dos provedores por nível de importância

| Provedor | Níveis de Importância | | | |
|----------|-----------------------|-------|--------|-------|
| | Essencial | Alto | Médio | Baixo |
| P_1 | 0,71 | 1,00 | 0,50 | 1 |
| P_3 | 0,85 | 0,35 | 0,7375 | 1 |
| P_4 | 0,77 | 0,485 | 0,50 | 0 |
| P_5 | 0,85 | 0,35 | 0,925 | 1 |

A seguir (Etapa 3), calcula-se a pontuação final e consequentemente o *ranking* para cada um dos 4 provedores concorrentes, de acordo com a Equação 7. Os valores dos coeficientes utilizados para a média ponderada são os valores apresentados na Tabela

2. Tais pesos são aplicados à pontuação de cada nível de importância calculados anteriormente (Tabela 7). A Tabela 8 apresenta a pontuação final dos Provedores 1, 3, 4 e 5. Assim, de acordo com a Tabela 8, é possível ordenar os quatro provedores candidatos em ordem decrescente de pontuação:

Tabela 8. Cálculo da pontuação final com 4 níveis de importância

| Fórmula geral: | |
|---|--|
| $P(i) = 0,5255 * P_{Essencial}(i) + 0,2825 * P_{Alto}(i) + 0,1413 * P_{Medio}(i) + 0,0508 * P_{Baixo}(i)$ | |
| Prov. | Pontuação final |
| P_1 | $0,5255 * (0,71) + 0,2825 * (1,00) + 0,1413 * (0,50) + 0,0508 * (1,00) = 0,7771$ |
| P_3 | $0,5255 * (0,85) + 0,2825 * (0,35) + 0,1413 * (0,7375) + 0,0508 * (1,00) = 0,7006$ |
| P_4 | $0,5255 * (0,77) + 0,2825 * (0,485) + 0,1413 * (0,50) + 0,0508 * (0) = 0,6123$ |
| P_5 | $0,5255 * (0,85) + 0,2825 * (0,35) + 0,1413 * (0,925) + 0,0508 * (1,00) = 0,7271$ |

1. P_1 com 0,7771 pontos;
2. P_5 com 0,7271 pontos;
3. P_3 com 0,7006 pontos; e
4. P_4 com 0,6123 pontos.

Logo, o Provedor 1 seria o mais apropriado para o usuário desse exemplo. O método pode retornar ao usuário uma lista contendo os w provedores melhor ordenados. Considerando $w = 3$, o retorno seria: $[\{1, P_1, 0.7771\}, \{2, P_5, 0.7271\}, \{3, P_3, 0.7006\}]$.

5. Conclusão

Esse trabalho especificou um método auxiliar à tomada de decisão que pontua e ordena, visando a seleção, o(s) mais apropriado(s) provedor(es) de nuvem, dentre um conjunto inicial de provedores, com base na requisição do usuário. A requisição deve apresentar os PIs de interesse, as preferências de valores para os mesmos e suas tolerâncias e a importância de um sobre o outro. O método é dividido em três etapas e separa os PIs em dois tipos: essenciais e não essenciais. Os não essenciais possuem diferentes níveis de importância. A pontuação final leva em consideração essa importância, resultando em um número real entre 0 e 1. Quanto mais perto de 1 mais adequado e preferível é o provedor em relação aos seus concorrentes para aquele usuário em questão. Um exemplo de aplicação do método foi apresentado, demonstrando sua utilização e a praticidade de adoção.

O principal benefício do método proposto é a sua alta generalidade e dimensionalidade, ou seja, capacidade de trabalhar com todo e qualquer tipo de PI disponível independentemente se o mesmo for quantitativo ou qualitativo. Ele apresenta como contribuição levar em conta tolerâncias para os valores quantitativos ou qualitativos desejados, baseadas na função de utilidade de cada PI, para a pontuação de cada nível de importância. Ainda, disponibiliza a possibilidade de cálculo de pesos para os níveis de importância baseado em uma técnica de avaliação pareada de critérios, que é a matriz de julgamentos, bastante utilizada em problemas de auxílio à tomada de decisão. O método também é simples e intuitivo, pois não exige um domínio ou modelagem matemática sofisticada para entendê-lo ou para utilizá-lo. O método é agnóstico, não restringindo quais PIs o usuário deve utilizar para selecionar seu provedor. Uma interface especializada pode ser

disponibilizada para o usuário, auxiliando-o a construir a sua expressão de entrada. A interface informa quais os PIs cadastrados que podem ser utilizados (explicando o que ele representa, o seu tipo, comportamento e domínio), quantos provedores estão disponíveis para a escolha, quais níveis de importância disponíveis para uso e que valor será atribuído entre eles.

Trabalhos futuros incluem a aplicação do método em cenários reais, bem como a criação de um corretor (*broker*) que incorpore o método desenvolvido e auxilie um conjunto diverso de usuários. Ainda, o refinamento e incorporação de automatização na definição dos valores das constantes de ponderação para o cálculo da pontuação de PIs quantitativos e qualitativos é uma tarefa planejada. Aplicar o método a instâncias consideravelmente grandes, assim como modelar e especificar uma maneira prática de obter os diversos PIs dos provedores de computação em nuvem e mantê-los atualizados é uma tarefa futura a ser desenvolvida.

Referências

- Achar, R. and Thilagam, P. (2014). A broker based approach for cloud provider selection. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1252–1257.
- Baranwal, G. and Vidyarthi, D. P. (2014). A framework for selection of best cloud service provider using ranked voting method. In *Advance Computing Conference (IACC), 2014 IEEE International*, pages 831–837.
- CSMIC (2014). Service measurement index framework. Technical report, Carnegie Mellon University, Silicon Valley, Moffett Field, California. Accessed in November 2016.
- Garg, S. K., Versteeg, S., and Buyya, R. (2013). A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 29(4):1012–1023.
- Hogan, M. D., Liu, F., Sokol, A. W., and Jin, T. (2013). *Nist Cloud Computing Standards Roadmap*. NIST Special Publication 500 Series. accessed in September 2016.
- Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. John Wiley & Sons, Littleton, Massachusetts.
- Saaty, T. L. (2004). Decision making – the analytic hierarchy and network processes (ahp/anp). *Journal of Systems Science and Systems Engineering*, 13:1–35.
- Shirur, S. and Swamy, A. (2015). A cloud service measure index framework to evaluate efficient candidate with ranked technology. *International Journal of Science and Research*, 4(3):1957–1961.
- Sundareswaran, S., Squicciarini, A., and Lin, D. (2012). A brokerage-based approach for cloud service selection. In *2012 IEEE Fifth International Conference on Cloud Computing*, pages 558–565.
- Wagle, S., Guzek, M., Bouvry, P., and Bisdorff, R. (2015). An evaluation model for selecting cloud services from commercially available cloud providers. In *7th International Conference on Cloud Computing Technology and Science*, pages 107–114.
- Zhang, Q., Cheng, L., and Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1(1):7–18.