

# Ranqueamento de Recursos na Internet das Coisas

## Explorando Algoritmos MCDA

Juan Burtet, Huberto K. Filho, Renato Dilli,  
Felipe C. Gruendemann, Ana Marilza Pernas, Adenauer Yamin

<sup>1</sup>Laboratory of Ubiquitous and Parallel Systems – UFPEL  
Pelotas, RS - Brasil

{jrburtet, hkaiser, renato.dilli, fcgruendemann, marilza, adenauer}  
@inf.ufpel.edu.br

**Resumo.** *A IoT caracteriza-se por possuir uma grande quantidade de dispositivos conectados em rede. O procedimento de descoberta e seleção de recursos neste cenário representa um desafio de pesquisa em aberto. Com o emprego do gerador e classificador de recursos concebidos, foi possível avaliar o desempenho do processo de ranqueamento, considerando uma grande quantidade de recursos. Os resultados com o algoritmo MCDA proposto são promissores.*

### 1. Introdução

A Internet das Coisas (IoT - *Internet of Things*), caracteriza-se pela presença de uma grande quantidade de objetos que pelo seu perfil operacional realizam conexões transientes na Internet, totalizando uma grande quantidade de recursos disponibilizados [Perera 2017]. Atualmente mais de seis bilhões de coisas estão conectadas à Internet disponibilizando serviços aos clientes e há previsão de 100 bilhões até 2025 [Gartner 2015] [BIS Research 2017].

Os recursos podem ser constituídos por *hardwares* heterogêneos, tais como, sensores e atuadores. O acesso aos dispositivos geralmente é realizado através de um ou mais serviços, caracterizando recursos de diferentes naturezas. Um desafio a ser vencido após a identificação e localização destes recursos é ranquear os serviços oferecidos para selecionar o que melhor atende a solicitação do usuário. Os processos de ranqueamento se concentram nas preferências do usuário, que frequentemente estabelecem uma ordem baseada na Qualidade de Serviço (QoS).

A Análise de Decisão de Múltiplos Critérios (MCDA - *Multiple Criteria Decision Analysis*) refere-se à tomada de decisões na presença de critérios múltiplos, geralmente conflitantes. Os algoritmos MCDA visam auxiliar no julgamento da tomada de decisão utilizando um conjunto de objetivos e critérios, estimando seus pesos de importância relativa e estabelecendo a contribuição de cada opção em relação a cada critério de desempenho. MCDA não é apenas um conjunto de teorias, metodologias e técnicas, mas também uma perspectiva específica para lidar com problemas de tomada de decisão [Figueira et al. 2005].

Este trabalho apresenta abordagens para geração e ranqueamento de recursos no contexto da IoT, com objetivo de avaliar o desempenho do algoritmo MCDA proposto. O restante do trabalho está organizado da seguinte forma: na Seção 2 é apresentado o método de ranqueamento de recursos utilizado, na Seção 3 é apresentado tanto um gerador,

como um classificador a ser utilizado para o ranqueamento de recursos, e na Seção 4 são feitas as considerações finais.

## 2. Modelo para Ranqueamento de Recursos Proposto

O processo de descoberta de recursos engloba a seleção dos recursos mais adequados à requisição do cliente. Nesta seção é apresentado o algoritmo MCDA proposto para o processo de ranqueamento de recursos utilizado neste trabalho.

O algoritmo MCDA proposto é uma adaptação dos algoritmos *The Simple Additive Weighting* (SAW) [Tzeng and Huang 2011] e *Web Service Relevancy Function* (WsRF) [Al-Masri and Mahmoud 2007], com a primeira etapa de normalização de dados proposta por [Liu et al. 2004].

Para a normalização da matriz são definidos dois vetores. O primeiro  $N = \{n_1, n_2, \dots, n_m\}$ . O valor de  $n_j$  pode ser 0 ou 1. Será 1 quando o aumento de  $q_{i,j}$  beneficia a requisição do cliente e 0 quando o aumento de  $q_{i,j}$  não beneficia a requisição do cliente. O segundo vetor  $C = \{c_1, c_2, \dots, c_m\}$  contém constantes com o máximo valor normalizado para cada atributo.

As seguintes etapas devem ser realizadas para o cálculo da avaliação dos recursos através do algoritmo MCDA:

1. Normalizar a matriz  $Q = (q_{ij})_{n \times m}$  de acordo com a Eq.(1) se o critério deve ser maximizado ou Eq.(2) se o critério deve ser minimizado. Nestas equações,  $\frac{1}{n} \sum_{i=1}^n q_{i,j}$  é a média dos atributos de qualidade  $j$  na matriz  $Q$  [Liu et al. 2004].

$$v_{i,j} = \begin{cases} \frac{q_{i,j}}{\frac{1}{n} \sum_{i=1}^n q_{i,j}} & \text{if } \frac{1}{n} \sum_{i=1}^n q_{i,j} \neq 0 \\ & \text{and } \frac{q_{i,j}}{\frac{1}{n} \sum_{i=1}^n q_{i,j}} < c_j \\ & \text{and } n_j = 1 \\ c_j & \text{if } \frac{1}{n} \sum_{i=1}^n q_{i,j} = 0 \\ & \text{and } n_j = 1 \\ & \text{or } \frac{q_{i,j}}{\frac{1}{n} \sum_{i=1}^n q_{i,j}} \geq c_j \end{cases} \quad v_{i,j} = \begin{cases} \frac{\frac{1}{n} \sum_{i=1}^n q_{i,j}}{q_{i,j}} & \text{if } q_{i,j} \neq 0 \\ & \text{and } n_j = 0 \\ & \text{and } \frac{\frac{1}{n} \sum_{i=1}^n q_{i,j}}{q_{i,j}} < c_j \\ c_j & \text{if } q_{i,j} = 0 \\ & \text{and } n_j = 0 \\ & \text{or } \frac{\frac{1}{n} \sum_{i=1}^n q_{i,j}}{q_{i,j}} \geq c_j \end{cases} \quad (1) \quad (2)$$

2. Calcular o vetor pontuação de cada opção disponível. Cada pontuação pode ser calculada usando a Eq.(3a) e o operador  $\max(v_j)$  representando o maior valor de atributo normalizado na coluna  $j$ . Portanto, precisamos definir uma matriz que represente a contribuição de pesos para cada recurso, onde  $w = \{w_1, w_2, w_3, \dots, w_j\}$ . Cada peso nesta matriz representa o grau de importância ou fator de peso associado a uma propriedade QoS específica. Os valores de desses pesos variam de 0 a 1. A Eq.(3b) soma todos os atributos de qualidade para o recurso  $R_i$ , onde  $N$  representa o número de atributos.

$$h_{i,j} = w_j \left[ \frac{v_{i,j}}{\max(v_j)} \right]; \quad R_i = \sum_{j=1}^N h_{i,j} \quad \text{e} \quad MCD(A_i) = \left[ \frac{100 * R_i}{\max(R)} \right]. \quad (3)$$

### 3. Prototipação e Testes

Com objetivo de analisar o desempenho do modelo de ranqueamento proposto, foi concebido um gerador de recursos na linguagem Python, adotada pelo grupo de pesquisa, que produz utilizando a biblioteca *random*, um conjunto de recursos aleatórios. Para cada atributo é selecionado um valor aleatório entre os valores mínimo e máximo definidos pelo seus atributos. Registre-se que estes valores mínimo e máximo tem por base o dataset [Al-Masri and Mahmoud 2007], que traduz um cenário real. Ao fim da criação do conjunto, é feito o ranqueamento de todos os recursos gerados.

As Tabelas 1, 2 e 3 apresentam o processo de ranqueamento de dez recursos produzidos pelo gerador, e após processados pelo classificador de recursos proposto.

| Tabela 1. Exemplo de Dataset |    |      |    |        | Tabela 2. Atributos Normalizados |      |      |      |       | Tabela 3. Atributos Classificados |      |      |      |      |     |
|------------------------------|----|------|----|--------|----------------------------------|------|------|------|-------|-----------------------------------|------|------|------|------|-----|
| RT                           | AV | TH   | RE | LA     | RT                               | AV   | TH   | RE   | LA    | RT                                | AV   | TH   | RE   | LA   | R   |
| 302,75                       | 89 | 7,1  | 73 | 187,75 | 1,70                             | 1,07 | 0,97 | 1,06 | 0,21  | 0,24                              | 0,91 | 0,33 | 0,94 | 0,00 | 61  |
| 482,00                       | 85 | 16   | 73 | 1      | 1,07                             | 1,02 | 2,19 | 1,06 | 38,56 | 0,15                              | 0,87 | 0,75 | 0,94 | 0,70 | 86  |
| 3321,4                       | 89 | 1,4  | 73 | 2,6    | 0,15                             | 1,07 | 0,19 | 1,06 | 14,83 | 0,02                              | 0,91 | 0,07 | 0,94 | 0,27 | 56  |
| 126,17                       | 98 | 12   | 67 | 22,77  | 4,07                             | 1,18 | 1,64 | 0,98 | 1,69  | 0,57                              | 1,00 | 0,56 | 0,86 | 0,03 | 76  |
| 107,00                       | 87 | 1,9  | 73 | 58,33  | 4,80                             | 1,04 | 0,26 | 1,06 | 0,66  | 0,67                              | 0,89 | 0,09 | 0,94 | 0,01 | 66  |
| 107,57                       | 80 | 1,7  | 67 | 18,21  | 4,78                             | 0,96 | 0,23 | 0,98 | 2,12  | 0,67                              | 0,82 | 0,08 | 0,86 | 0,04 | 62  |
| 255,00                       | 98 | 1,3  | 67 | 40,8   | 2,02                             | 1,18 | 0,18 | 0,98 | 0,95  | 0,28                              | 1,00 | 0,06 | 0,86 | 0,02 | 56  |
| 136,71                       | 76 | 2,8  | 60 | 11,57  | 3,76                             | 0,91 | 0,38 | 0,87 | 3,33  | 0,53                              | 0,78 | 0,13 | 0,77 | 0,06 | 57  |
| 102,62                       | 91 | 15,3 | 67 | 0,93   | 5,00                             | 1,09 | 2,10 | 0,98 | 41,46 | 0,70                              | 0,93 | 0,72 | 0,86 | 0,75 | 100 |
| 200,00                       | 40 | 13,5 | 67 | 41,66  | 2,57                             | 0,48 | 1,85 | 0,98 | 0,93  | 0,36                              | 0,41 | 0,63 | 0,86 | 0,02 | 58  |

A primeira etapa da classificação é a normalização dos dados. Para tanto, consideramos os vetores  $N=\{0,1,1,1,0\}$  e  $C=\{5,2,3,2,50\}$ . Os atributos *Response Time* e *Latency* qualificam o recurso com valores baixos obtidos pela Eq.(1) e *Availability*, *Throughput* e *Reability* qualificam com valores altos pela Eq.(2).

A Tabela 3 apresenta os valores dos atributos após aplicar a Eq.(3)a, ou seja dividir o valor normalizado da Tabela 2 pelo maior valor normalizado de cada coluna, multiplicado pelos pesos definidos pelo especialista  $w=\{0.70,1.00,0.75,0.94,0.75\}$ . O valor MCDA é obtido aplicando a Eq.(3)b, que irá somar todos os valores de atributos em cada linha. Após é aplicada a Eq.(3)c que irá qualificar o recurso com um valor que vai de 0 a 100. Será dado o valor 100 para o melhor recurso do *dataset*. O resultado da classificação está informado na coluna "R". A classificação final é apresentada ao cliente em ordem decrescente, ou seja, do recurso mais qualificado ao menos qualificado.

Com o emprego do *software* desenvolvido para geração de recursos, foram criados 4 conjuntos com seus atributos de qualidade. Para o processo de criação destes recursos, optou-se por utilizar conjuntos com 3, 5 e 7 atributos com o objetivo de apontar a diferença de tempo no cálculo MCDA considerando diferentes quantidades de atributos. Além disso, foram criados *datasets* de tamanhos diferentes para analisar o aumento de tempo para conjuntos com diferentes quantidades de recursos e com mesmo número de atributos. O tamanho dos *datasets* testados foram de [10.000; 100.000; 500.000;

1.000.000] recursos. O desempenho em cada um dos testes é visto na Tabela 4 (Tempo de criação de *datasets*) e na Tabela 5 (Tempo de classificação MCDA)

**Tabela 4. Tempo de geração dos recursos e seus atributos**

| Atrib | Recursos (x 1.000) |        |        |         |
|-------|--------------------|--------|--------|---------|
|       | 10                 | 100    | 500    | 1.000   |
| 3     | 0.86s              | 8.69s  | 45.87s | 88.17s  |
| 5     | 1.30s              | 13.92s | 65.38s | 130.17s |
| 7     | 1.77s              | 17.40s | 87.37s | 174.49s |

**Tabela 5. Tempo de ranqueamento dos recursos**

| Atrib | Recursos (x 1.000) |       |       |        |
|-------|--------------------|-------|-------|--------|
|       | 10                 | 100   | 500   | 1.000  |
| 3     | 0.10s              | 1.01s | 5.15s | 10.08s |
| 5     | 0.15s              | 1.44s | 7.20s | 14.50s |
| 7     | 0.18s              | 1.88s | 9.59s | 18.79s |

Todos os testes foram feitos em uma máquina com Intel I5-3570K CPU @ 3.40 GHz e 4.00 GB de RAM. Com uma análise dos resultados, foi notado um ótimo desempenho no algoritmo SAW para classificação de recursos. Também foi notado que houve um aumento linear no tempo de geração e ranqueamento, conforme o aumento do número de recursos.

#### 4. Considerações Finais

A utilização do algoritmo MCDA proposto para a classificação de recursos no contexto da Internet das Coisas apresentou um ótimo desempenho, considerando a literatura na área [Al-Masri and Mahmoud 2007], mesmo com um grande aumento no conjunto de dados e no número de atributos de qualidade.

Em trabalhos futuros, novos testes serão feitos, comparando a acurácia do ranqueamento obtido pelo algoritmo MCDA com novos métodos de ranqueamento, tais como Lógica *Fuzzy* e Aprendizagem de Máquina.

#### Referências

- Al-Masri, E. and Mahmoud, Q. H. (2007). QoS-based discovery and ranking of Web services. In *Proceedings - International Conference on Computer Communications and Networks, ICCCN*, pages 529–534.
- BIS Research (2017). Global Sensors in Internet of Things (IoT) Devices Market, Analysis & Forecast: 2016 to 2022. Technical report.
- Figueira, J., Greco, S., and Ehrgott, M. (2005). Multiple criteria decision analysis: state of the art surveys. *Multiple Criteria Decision Analysis State of the Art Surveys*, 78.
- Gartner (2015). Gartner says 6.4 billion connected "things" will be in use in 2016, up 30 percent from 2015.
- Liu, Y., Ngu, A. H., and Zeng, L. Z. (2004). QoS computation and policing in dynamic web service selection. *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 66–73.
- Perera, C. (2017). *Sensing as a Service for Internet of Things: A Roadmap*. Leanpub Publishers.
- Tzeng, G. H. and Huang, J. J. (2011). *Multiple Attribute Decision Making: Methods and Applications*. A Chapman & Hall book. Taylor & Francis.