

IoT-DFR³: Explorando a Geração Dinâmica de Regras Fuzzy para Classificação de Recursos na IoT

Patrick Fernandes, Renato Dilli, Carlos Silva, Renata Reiser e Adenauer Yamin

Resumo— Com o crescente número de recursos na IoT, surge o desafio de selecionar o recurso disponível que ofereça melhor Qualidade de Serviço (QoS) ao cliente, considerando suas preferências. Este trabalho tem como objetivo a concepção da proposta IoT-DFR³, que contempla: (i) emprego de Lógica Fuzzy Tipo-2 Intervalar para tratamento de incertezas na definição dos atributos de QoS; (ii) geração dinâmica de Regras Fuzzy, explorando o método de Wang-Mendel; e (iii) tradução das preferências dos clientes em regras Fuzzy. Além disso, um cenário de uso, comparando a classificação de recursos, aplicando diferentes preferências de cliente é apresentado.

Palavras-Chave— Geração de Regras Fuzzy; Classificação de Recursos; Internet das Coisas; Lógica Fuzzy Tipo-2.

Abstract— With the growing number of resources in the IoT, the challenge arises to select the available resource that offers the best Quality of Service (QoS) to the client, considering their preferences. This work aims to conceive the IoT-DFR³ proposal, which contemplates: (i) use of Interval Type-2 Fuzzy Logic to deal with uncertainties in the definition of QoS attributes; (ii) dynamic generation of Fuzzy Rules, exploring the Wang-Mendel method; and (iii) translating client preferences into Fuzzy rules. In addition, a usage scenario, comparing the classification of resources, applying different client preferences is presented.

Keywords— Generating Fuzzy Rules; Resource Classification; Internet of Things; Type-2 Fuzzy Logic.

I. INTRODUÇÃO

A Internet das Coisas (IoT) é considerada uma evolução computacional a qual contempla uma infinidade de objetos físicos embarcados com sensores e atuadores, conectados por redes com e sem fio e que se comunicam através da Internet [1]. A previsão é que até 2030 em torno de 125 bilhões de recursos estejam conectados à IoT [2], podendo cada um destes recursos oferecer mais de um serviço. Deste modo, a seleção dos recursos cujos serviços melhor atendam os requisitos de um cliente, entre os vários recursos aptos, pode resultar em uma tarefa complexa.

Atendendo o perfil operacional da infraestrutura da IoT, uma proposta de seleção, deve considerar os requisitos funcionais e os não funcionais dos recursos envolvidos. Os requisitos funcionais indicam aquilo que os serviços devem oferecer. Os requisitos não funcionais, por sua vez, são associados à

Qualidade de Serviço (QoS), apontam para as expectativas operacionais dos serviços. Por exemplo, a “disponibilidade”, que é a porcentagem de tempo que um serviço permanece funcionando; e/ou o “tempo de resposta”, que é o tempo necessário para um serviço responder [3].

Os requisitos funcionais são utilizados para definir se um certo recurso é capaz de atender as demandas de um cliente, por sua vez, a avaliação de parâmetros não funcionais de QoS dos serviços fornecidos pelos recursos, estabelecem a base para selecionar entre os recursos apropriados, quais são os mais aptos para atender a demanda de um cliente.

Para tanto, é preciso classificar os recursos segundo os parâmetros não funcionais, para então selecionar os mais adequados. Nesta classificação, se faz necessário considerar as incertezas dos clientes na caracterização de suas preferências, quanto aos parâmetros de QoS que lhe são mais apropriados.

A expectativa deste trabalho é utilizar Lógica Fuzzy (LF) Tipo-2 Intervalar para tratamento destas incertezas dos clientes, o que se torna bastante complexo de ser feito com as lógicas clássicas. A LF procura representar o conhecimento e o raciocínio humano de modo o mais real possível, facilitando assim o desenvolvimento de sistemas complexos [4].

Por outro lado, os aspectos de heterogeneidade e escalabilidade da IoT, fazem com que o número de regras a serem utilizadas na LF cresça, a medida que aumenta o número de parâmetros de QoS considerados. Nesta perspectiva, surge o desafio de pesquisa deste trabalho, que é reduzir o esforço associado a criação manual de regras para o sistema Fuzzy empregado na classificação de recursos.

Neste sentido, o objetivo geral deste trabalho é conceber uma proposta denominada IoT-DFR³ (*IoT-Dynamic Fuzzy Rules for Resource Ranking*), para geração dinâmica de Regras Fuzzy, explorando o método de Wang-Mendel (WM) [5]. O IoT-DFR³ foi desenvolvido, enquanto uma arquitetura de software, com o intuito de ser integrado à proposta EXEHDA-RR [6], direcionada à seleção de recursos na IoT.

Existem vários métodos para geração de regras na LF. Dentre eles, optamos pelo emprego do método WM que é oportuno para construção de regras Fuzzy, considerando dados de natureza numérica, os quais representam amostras de um determinado conjunto de dados [7]. Além disto, esse método fornece precisão ao elaborar as regras, e as produz de forma que possa ser manipuladas por não especialistas [8].

Este trabalho está organizado do seguinte modo: na Seção 2 são apontados os trabalhos relacionados, na Seção 3, a proposta IoT-DFR³, na Seção 4, o cenário de uso e, por fim, na Seção 5 as considerações finais.

Patrick Jean Barboza Fernandes, Programa de Pós-graduação em Engenharia Eletrônica e Computação, Universidade Católica de Pelotas-RS, e-mail: patrick.fernandes@sou.ucpel.edu.br; Renato Marques Dilli, Instituto Federal Sul-rio-grandense, e-mail: renatodilli@ifsul.edu.br; Carlos Vinicius Machado Silva, Universidade Federal do Rio Grande do Sul, e-mail: carlosvinicius.ms@gmail.com; Renata Hax Sander Reiser, Universidade Federal de Pelotas, e-mail: reiser@inf.ufpel.edu.br; Adenauer Corrêa Yamin, Universidade Católica de Pelotas, e-mail: adenauer.yamin@ucpel.edu.br. Este trabalho foi parcialmente financiado pela CAPES, CNPq e FAPERGS.

II. TRABALHOS RELACIONADOS

Uma revisão bibliográfica foi realizada, sendo apontados nesta Seção somente os trabalhos que mais se relacionam ao IoT-DRF³. Esta revisão contribuiu para as discussões referentes as funcionalidades a serem propostas para a arquitetura de software do IoT-DRF³.

O trabalho [9] aponta um modelo baseado em Lógica Fuzzy com o intuito de ajudar os consumidores de serviços da Web a classificar o serviço mais adequado, tendo em conta suas preferências.

A proposta [10], aborda o uso de Lógica Fuzzy para seleção de recursos, composição e adaptação de QoS em tempo de execução. Utiliza atributos de QoS como entrada de um Sistema Baseado em Regras Fuzzy (SBRF). O sistema implementa um conjunto de mecanismos de adaptação com base no monitoramento da saída e na nova seleção automática de serviços.

Em [11] é abordado um modelo baseado em Lógica Fuzzy para seleção de serviços em mercado eletrônico de acordo com as preferências dos clientes. A proposta, denominada de FOCUSS, inclui os métodos Fuzzy AHP para obter os pesos dos atributos de QoS, Fuzzy DM para obter os valores de QoS desejados pelo usuário e por fim, um módulo de Fuzzy MOP para avaliar os serviços com relação as preferências do usuário.

O trabalho [12], indica um modelo Fuzzy para seleção de serviço de nuvem, considerando os atributos de QoS. Empregando o método do AHP (processo de hierarquia analítica) para calcular os pesos e Fuzzy TOPSIS (preferência de ordem por similaridade à solução ideal) para a classificação final.

Ao analisar as contribuições destes trabalhos e considerando que esta pesquisa tem como foco central propor uma arquitetura de software, IoT-DRF³, para geração de regras Fuzzy e que permita o cliente selecionar de modo confortável suas preferências para cada atributo de QoS, oferecendo suporte a eventuais incertezas no momento desta definição, destacá-amos que a geração automatizada de regras constitui o principal diferencial deste trabalho.

III. IoT-DRF³: ARQUITETURA E FUNCIONALIDADES

Esta seção apresenta a concepção da proposta IoT-DRF³. A premissa central buscada é gerar dinamicamente as Regras Fuzzy que serão empregadas para classificação de recursos na IoT, visando também a integração da arquitetura de software do IoT-DRF³ ao EXEHDA-RR.

A. Explorando a Geração Automática de Regras Fuzzy

A adoção do Método de WM na proposta IoT-DRF³, proporcionará agilidade ao gerar as regras Fuzzy de forma automática e contribuirá com a redução no número de regras. Por exemplo, ao utilizar 3 atributos, teríamos um total de 27 regras. Agora, aplicando o Método de WM para os mesmos 3 atributos, consegue-se uma redução superior a 50% no número total dessas mesmas regras. Lembrando que quanto maior o número de atributos, conseqüentemente maior o número de regras.

Além do Método de WM, é permitido a inserção manual de regras na Base de Regras Fuzzy. Possibilitando assim, acrescentar um conjunto de regras específicas para certas preferências. Com isso, o IoT-DRF³, aprimorará o funcionamento do EXEHDA-RR no desenvolvimento e na manipulação das regras.

B. Explorando Preferências do Especialista

O cliente define quais são suas prioridades por meio de variáveis linguísticas e um especialista pode atribuir regras adicionais para contemplar essas preferências

O especialista também desenvolve as funções de pertinência referentes à Lógica Fuzzy Tipo-2 Intervalar para cada atributo presente no conjunto de preferências que está sendo considerado.

A proposta IoT-DRF³ permite que o cliente selecione essas preferências de cada um dos atributos de QoS dos recursos. Esses atributos possuem as opções de configuração *High* (H), *Medium* (M) ou *Low* (L).

Os atributos de QoS dos recursos são normalizados e então, submetidos como entrada no Sistema Fuzzy Tipo-2, sendo fuzzificados através de funções de pertinência triangulares e trapezoidais. A arquitetura da proposta IoT-DRF³ é exibida na Figura 1 e suas etapas são caracterizadas a seguir.

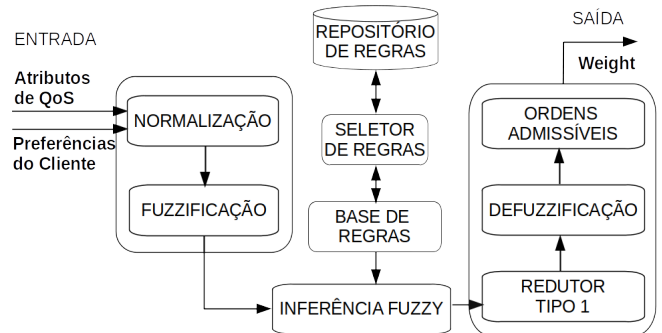


Fig. 1. Arquitetura de Software da Proposta IoT-DRF³

C. Normalização e Fuzzificação

No processo de normalização, o melhor valor de atributo de QoS é normalizado em 100. Esse procedimento é feito para cada um dos atributos. Porém, os atributos que já têm os seus valores definidos numa escala entre 0 e 100, não são normalizados, como acontece com o atributo AV. A normalização é executada conforme a Eq.(1) se o critério deve ser maximizado ou Eq.(2) se o critério deve ser minimizado.

$$h_{i,j} = \left[\frac{v_{i,j} * 100}{\max(v_j)} \right] \quad (1) \quad h_{i,j} = \left[\frac{\min(v_j) * 100}{v_{i,j}} \right] \quad (2)$$

Ao finalizar a normalização, é realizado o processo de fuzzificação, que transforma um valor crisp em um valor Fuzzy, \tilde{A} apontando o grau de pertinência do elemento $x \in U$ no conjunto Fuzzy A , geralmente dado na forma trapezoidal ($\tilde{A} = (a, b, c, d)$) ou triangular ($\tilde{A} = (a, b, c)$) [13].

No sistema Fuzzy Tipo-2 intervalar, cada atributo é submetido como entrada. Foram utilizados três atributos de entrada cujos nomes são *Response Time* (RT), *Availability* (AV) e *Throughput* (TH). E como variável de saída, somente o peso para o recurso avaliado, definido de *Weight*.

Os valores dos Termos Linguísticos H, M e L para as variáveis RT, AV e TH, são exibidos na Tabela I. Ao qual, o Termo Linguístico H foi determinado com função de pertinência triangular e os Termos M e L foram atribuídos com funções de pertinência trapezoidal. Na coluna do meio e na coluna da direita, indicam os limites superiores e inferiores, *FP Superior* e *FP Inferior*, respectivamente.

TABELA I

IoT-DFR³ - VARIÁVEIS LINGUÍSTICAS TIPO-2: RT, AV E TH

Termo Linguístico	FP Superior	FP Inferior
H - High	(56,100,100)	(68,100,100)
M - Medium	(31,46,68,83)	(43,58,56,71)
L - Low	(0,0,38,58)	(0,0,26,46)

A variável *Weight*, de saída, foi definida utilizando funções de pertinência trapezoidal, cujos valores são representados na Tabela II.

TABELA II

IoT-DFR³ - VARIÁVEL LINGUÍSTICA TIPO-2: WEIGHT

Termo Linguístico	FP Superior	FP Inferior
H - High	(46,76,100,100)	(58,88,100,100)
M - Medium	(6,36,63,93)	(18,48,51,81)
L - Low	(0,0,28,53)	(0,0,16,41)

D. Repositório, Seletor e Base de Regras

Os conjuntos de regras empregados, tanto os gerados pelo método WM, quanto os adicionados manualmente, são armazenados no Repositório de Regras, assim como as funções de pertinência desenvolvida pelo especialista. Com isso, o Seletor de Regras tem o objetivo de filtrar o conjunto que será utilizado no processo de Classificação, de acordo com as preferências estabelecidas pelo cliente.

Já a Base de Regras, aponta como as variáveis são executadas em Termos Linguísticos pelo Motor de Inferência. Conforme o controlador Fuzzy concebido por Mamdani [14], foi desenvolvido um método para o processo de decisão baseado em regras do tipo “*se X então Y*”, no qual tanto o antecedente quanto o conseqüente são valores de variáveis linguísticas, expressos por meio de Conjuntos Fuzzy.

A intersecção e a união, são os operadores Fuzzy mais utilizados. O método de Mamdani agrega as regras por meio do operador de união (definido como “ou”) que é representado pelo operador máximo e, em cada regra, o operador de intersecção (definido como “e”) que é indicado pelo operador mínimo.

A proposta IoT-DFR³, também permite criar e configurar um perfil com diferentes regras e/ou outras funções de pertinência. A Figura 2 exibe a interface com essas opções.

Para a elaboração das regras, mesmo utilizando o Método de WM, aplica-se a lógica de que o menor valor dos Termos Linguísticos entre as variáveis de entrada, seja atribuída na

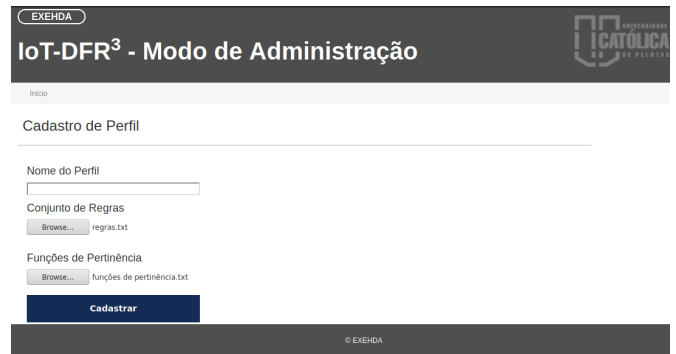


Fig. 2. Modo de Administração - Cadastro de Perfil

variável de saída. Por exemplo, a primeira regra da Tabela IV pode ser desmembrada do seguinte modo: *Se HighRT e HighAV e LowTH então LowWeight*. Ou seja, a variável de entrada *TH* recebe o menor termo linguístico (*Low*), então, a variável de saída *Weight* recebe o mesmo termo *Low*.

E. Motor de Inferência Fuzzy

O objetivo do Motor de inferência Fuzzy é executar as regras, conforme a Base de Regras, elaboradas com operadores de composição e conectivos lógicos da LF. Este bloco aplica o Método de Inferência de Mamdani para Conjuntos Fuzzy Tipo-2. E após concluir seu propósito, é necessário modelar o Conjunto Fuzzy Tipo-2 em Conjunto Fuzzy Tipo-1. Para isso, utiliza-se um redutor de tipo.

F. Redutor Tipo-1, Defuzzificação e Ordens Admissíveis

O Redutor Tipo-1 transforma o Conjunto Fuzzy Tipo-2 Intervalar em Tipo-1. Este bloco efetua o Método do centroide, que utiliza o algoritmo de Karnik-Mendel (KM) [15] para aplicar a modelagem. O Método fornece o intervalo fechado $[C_L, C_R]$ sendo o mínimo (C_L) e o máximo (C_R) dos centroides de Conjuntos Fuzzy Tipo-1, cuja função de pertinência está contida na mancha de incerteza (*Footprint of Uncertainty - FOU*) [16].

Nota-se que ao defuzzificar os intervalos para obter um valor crisp com o propósito de ordenar e para fazer a classificação final, em alguns casos, o valor defuzzificado de intervalos diferentes permanece o mesmo, o que dificulta no processo da classificação. Por isso, a defuzzificação não é efetuada como a média entre dois valores, onde o primeiro calcula o ponto final esquerdo do intervalo (C_L) e o último calcula o ponto final direito (C_R). Portanto, optou-se por preservar os intervalos e ordená-los por meio de Ordens Admissíveis.

Ao terminar esta etapa, obteve-se uma lista de intervalos, gerados a partir dos Conjuntos Fuzzy Tipo-2 Intervalar, com muitos elementos que podem não ser comparáveis pela ordem parcial usual, uma ordem total de intervalos apontados com base nas ordens admissíveis, como proposto por [17].

Nesta perspectiva, o desenvolvimento do IoT-DFR³ não restringe o uso de operadores de agregação (como média aritmética) aplicados sobre os limites superior e o limite dos dados do intervalo no processo final da etapa de defuzzificação.

Esse método modela as incertezas relacionadas com as preferências do cliente e a imprecisão dos cálculos computacionais. Deste modo, é possível garantir a comparação de todos os dados do intervalo de saída. Assim, considera a ordem de Xu e Yager [17], ao qual os recursos são classificados em ordem decrescente de intervalos através da ordem admissível.

IV. CENÁRIO DE USO

Este cenário de uso contempla requisições por recursos. Estes recursos inerentes à IoT, são disponibilizados pelo *QWS Dataset* versão 2.0 [18], contendo um total de 2.507 recursos, atualizados em novembro de 2019. O *Dataset* possui ao todo nove atributos de QoS e foram selecionados três destes, que são entendidos como os mais significativos para as demandas do cliente, o RT, AV e TH.

O IoT-DFR³ utiliza a ferramenta *Juzzy*, proposta por [19], para manipulação do Sistema Baseado em Regras Fuzzy Tipo-2 (SBRF2). Ambos foram desenvolvidos em *Java*, possibilitando a integração destas.

Foi elaborada uma avaliação para classificar 13 recursos (Tabela III), na qual foram geradas regras Fuzzy automaticamente, atribuindo as preferências do cliente para HHH, HLM e MLH com o propósito de comparar entre estas, qual melhor satisfaz as necessidades do cliente a fim de oferecer mais Qualidade de Serviço (QoS).

Para explicar, por exemplo, o significado da preferência HLM, denota-se que o atributo RT é *High*, o AV é *Low* e TH *Medium*.

A Figura 3 exibe as opções para seleção das preferências e o perfil de regras a ser selecionado. No caso, o “Default”, ao qual as regras são geradas pelo Método de WM.

A Tabela III, aponta os valores dos atributos antes de serem normalizados e após a normalização. Essa mesma entrada, com os atributos normalizados e a saída W indicada na Tabela III, é utilizada para gerar o conjunto de regras pelo Método de WM. Essa saída W é o consequente da regra, onde o valor 1 significa *High*, 2 *Medium* e 3 *Low*.

TABELA III
CENÁRIO DE USO - AMOSTRA COM 13 RECURSOS

ID	Original			Normalizado			W
	RT	AV	TH	RT	AV	TH	
01	221,48	90	10,9	99,77	90	98,20	1
02	239,22	85	7,7	92,37	85	69,37	2
03	261,09	85	5,4	84,63	85	48,65	3
04	256,39	85	5,1	86,18	85	45,95	3
05	220,96	87	11,1	100	87	100	1
06	243,91	85	7,1	90,59	85	63,96	2
07	223,83	92	10,8	98,72	92	97,30	1
08	250,74	86	7	88,12	86	63,06	2
09	246,43	85	9,7	89,66	85	87,39	1
10	277,35	86	6,3	79,67	86	56,76	2
11	280,39	86	6,4	78,80	86	57,66	2
12	275,87	85	6,5	80,10	85	58,56	2
13	267,43	86	5,7	82,62	86	51,35	2

A Tabela IV, exibe o conjunto de regras produzido pelo Método de WM e processadas pelo Motor de Inferência Fuzzy com a preferência HHH. A primeira coluna apresenta a regra completa, e na segunda coluna, os pesos das regras estabelecido pelo Método de WM. Já na Tabela V e Tabela VI,

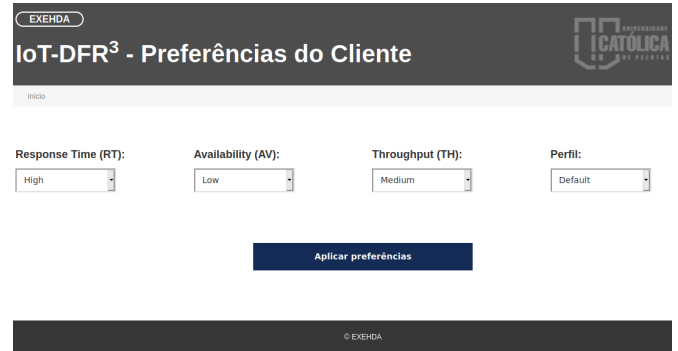


Fig. 3. Cenário de Uso - Preferências do Cliente (HLM)

apresentam as regras adicionais, aplicadas manualmente, em conjunto com as regras fixas (Tabela IV), para as preferências HLM e MLH respectivamente.

TABELA IV
IoT-DFR³ - BASE DE REGRAS MAMDANI (MÉTODO WM)

Regras	Pesos
Se HighRT e HighAV e LowTH então LowWeight	0.23
Se HighRT e HighAV e MediumTH então MediumWeight	0.40
Se HighRT e HighAV e HighTH então HighWeight	0.70

TABELA V
IoT-DFR³ - BASE DE REGRAS MAMDANI PARA (HLM)

Regras
Se HighRT então HighWeight
Se MediumTH então MediumWeight

TABELA VI
IoT-DFR³ - BASE DE REGRAS MAMDANI PARA (MLH)

Regras
Se MediumRT e HighTH então MediumWeight
Se HighRT HighWeight

Após o cliente clicar no botão “Aplicar preferências”, é apresentada a interface resultante da classificação para a preferência selecionada, HLM, Figura 4. A lista está ordenada pela ID do recurso e os valores dos atributos estão normalizados. O intervalo resultante da defuzzificação é caracterizado pelo Centroides Inferior e Superior e a última coluna exibe a classificação, onde o número 1 é o melhor recurso apto e 13 para o pior.

A Figura 5 apresenta a comparação da classificação dos 13 recursos para as preferências do cliente HHH, HLM e MLH. Nessa Figura, o eixo horizontal exibe o ID do recurso e o eixo vertical o resultado da classificação.

O melhor recurso para a preferência HHH foi o de ID 7, pois quanto maior o valor do atributo AV, melhor o resultado. Se comparar o recurso de ID 7 com o de ID 1, o ID 7 ainda é melhor mesmo tendo os valores de RT e TH maiores, pois o AV especificado em *High* possui maior prioridade.

Já para as preferências HLM e MLH, o melhor recurso foi o de ID 5. Pois recebem como entrada os atributos RT = 100, AV = 87 e TH = 100. Pelo atributo AV ser atribuído como

EXEHDA

IoT-DFR³ - Classificação para a Preferência HLM

ID	Response Time(RT)	Availability (AV)	Throughput (TH)	Centróide Inferior	Centróide Superior	Classificação
1	99,77	90,00	98,20	81,05	83,86	2
2	92,37	85,00	69,37	60,27	69,59	5
3	84,63	85,00	48,65	52,45	62,45	9
4	86,18	85,00	45,95	52,81	63,09	8
5	100,00	87,00	100,00	81,09	83,89	1
6	90,59	85,00	63,96	58,35	63,85	6
7	98,72	92,00	97,30	80,90	83,75	3
8	88,12	86,00	63,06	57,51	62,70	7
9	89,66	85,00	87,39	79,40	82,65	4
10	79,67	86,00	56,76	53,90	58,96	12
11	78,80	86,00	57,66	53,47	58,68	13
12	80,10	85,00	58,56	54,10	59,11	11
13	82,62	86,00	51,35	53,40	60,77	10

© EXEHDA

Fig. 4. Cenário de Uso - Classificação da Preferência HLM

Low em ambas preferências e os atributos RT e TH receberem o maior valor, o ID 5 é classificado como melhor recurso para essas duas preferências.

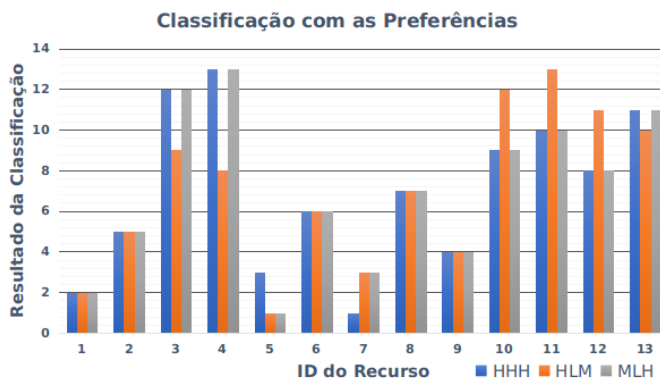


Fig. 5. Comparação dos resultados com preferências HHH, HLM, MLH

V. CONSIDERAÇÕES FINAIS

A tarefa de seleção de recursos na Internet das Coisas pode tornar-se complexa, se fazendo muitas vezes necessária a manipulação de uma grande quantidade de recursos, podendo o cliente também ter incertezas quanto aos atributos de QoS que irá considerar.

Neste trabalho, foi apresentada a arquitetura de software da proposta IoT-DFR³, cuja principal contribuição está na geração de regras na LF para classificação de recursos na IoT, considerando as preferências do cliente.

No cenário de uso, comparamos os resultados da classificação de um conjunto de 13 recursos com as preferências do cliente definida para HHH, HLM e MLH, e o perfil de configuração definido como “Default”. Resultando no melhor recurso de ID 7 para a preferência HHH e de ID 5 para HLM e MLH.

A proposta IoT-DFR³ permitiu gerar as regras Fuzzy de forma automática, empregando o método de WM. Apresentou uma redução significativa no número de regras, contribuindo assim, para uma melhor otimização da Base de Regras Fuzzy.

O IoT-DFR³ mostrou-se promissor, devendo ser integrado à proposta EXEHDA-RR para classificação de recursos na IoT.

Dentre os trabalhos futuros, destaca-se o desenvolvimento de uma interface gráfica para gerenciamento e visualização das regras Fuzzy.

REFERÊNCIAS

- [1] M. C. Schenfeld, L. Amaral, E. de Matos, and F. Hessel, “Arquitetura para fog computing em sistemas de middleware para internet das coisas,” in *Anais do XLIII Seminário Integrado de Software e Hardware*. Porto Alegre, RS, Brasil: SBC, 2016, pp. 199–209. [Online]. Available: <https://sol.sbc.org.br/index.php/semish/article/view/9530>
- [2] IHS Markit, “The Internet of Things: a movement, not a market,” p. 9, 2017.
- [3] P. A. Khutade and R. Phalnikar, “QoS Aware Web Service Selection and Ranking Framework Based on Ontology,” *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 4, no. 3, pp. 77–81, 2014.
- [4] M. Silva, M. A. Cardoso, M. C. Machado, and A. P. L. Ferreira, “Sistema de inferência fuzzy para estimativa de crescimento populacional,” *Anais do Salão Internacional de Ensino, Pesquisa e Extensão*, vol. 11, no. 2, 2019.
- [5] L. X. Wang and J. M. Mendel, “Generating Fuzzy Rules by Learning from Examples,” *IEEE Transactions on Systems, Man and Cybernetics*, 1992.
- [6] R. Dilli, A. Argou, A. Pernas, R. Reiser, and A. Yamin, “EXEHDA-RR : Uma proposta para tratar incertezas e otimizar o processo de classificação de recursos na IoT,” *Simpósio Brasileiro de Computação Ubíqua - CSBC*, 2018.
- [7] M. C. d. A. Silva, “Aplicação de um sistema fuzzy para classificação de opinião em diferentes domínios,” Ph.D. dissertation, Instituto de Matemática. Departamento de Ciência da Computação, 10 2015.
- [8] M. Santos and B. Mendoza, “Identificación borrosa de un cultivo experimental,” in *XXXIX Jornadas de Automática*. Área de Ingeniería de Sistemas y Automática, Universidad de Extremadura, Septiembre 2018, pp. 888–893.
- [9] H. Belouaar, O. Kazar, and N. Kabachi, “A new model for web services selection based on fuzzy logic,” *Courrier du Savoir*, vol. 1, no. 26, pp. 393–400, 2018.
- [10] A. K. Tripathy and P. K. Tripathy, “Fuzzy QoS requirement-aware dynamic service discovery and adaptation,” *Applied Soft Computing Journal*, vol. 68, no. November, pp. 136–146, 2018. [Online]. Available: <https://doi.org/10.1016/j.asoc.2018.03.038>
- [11] A. Ezenwoke, “Fuzzy Hybrid Approach for Ranking and Selecting Services in Cloud-based Marketplaces,” *Journal of Artificial Intelligence*, vol. 11, no. 1, pp. 9–17, 2018. [Online]. Available: <http://www.scialert.net/abstract/?doi=jai.2018.9.17>
- [12] R. R. Kumar, S. Mishra, and C. Kumar, “Prioritizing the solution of cloud service selection using integrated MCDM methods under Fuzzy environment,” *Journal of Supercomputing*, vol. 73, no. 11, pp. 4652–4682, 2017.
- [13] L. Barros and R. Bassanezi, *Tópicos de lógica fuzzy e biomatemática*. UNICAMP/IMECC, 2010.
- [14] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” *International Journal of Man-Machine Studies*, 1975.
- [15] J. M. Mendel, *Uncertain Rule-Based Fuzzy Systems: Introduction and New Directions*, 2nd ed. Springer International Publishing, 2017. [Online]. Available: <https://books.google.com.br/books?id=02UkDwAAQBAJ>
- [16] N. Cabrera, “Aplicação da Extensão de Zadeh para Conjuntos Fuzzy Tipo 2 Intervalar,” Ph.D. dissertation, Universidade Federal de Uberlândia, 2014.
- [17] Z. Xu and R. R. Yager, “Some geometric aggregation operators based on intuitionistic fuzzy sets,” *International Journal of General Systems*, vol. 35, no. 4, pp. 417–433, 2006.
- [18] E. Al-Masri and Q. H. Mahmoud, “Investigating web services on the world wide web,” in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 795–804.
- [19] C. Wagner, “Juzzy - A Java based toolkit for Type-2 Fuzzy Logic,” *Proceedings of the 2013 IEEE Symposium on Advances in Type-2 Fuzzy Logic Systems, T2FUZZ 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013*, vol. 1, no. April 2013, pp. 45–52, 2013.