Relevance of Using Interpretability Indexes for the Design of Schedulers in Cloud Computing Systems

Sebastián García Galán
Engineering Telecommunication Department
University of Jaén
Linares (Jaén), Spain
sgalan@ujaen.es

Mouad Seddiki
Engineering Telecommunication Department
University of Jaén
Linares (Jaén), Spain
ramdyt@hotmail.com

Rocio J. Perez de Prado
Engineering Telecommunication Department
University of Jaén
Linares (Jaén), Spain
rperez@ujaen.es

Enrique Muñoz Expósito
Engineering Telecommunication Department
University of Jaén
Linares (Jaén), Spain
jemunoz@ujaen.

Adam Marchewka
Institute of Telecommunications and Computer Sciences
University of Science and Technology
Bydgoszcz, Poland
adimar@utp.edu.pl

Nicolás Ruiz Reyes
Engineering Telecommunication Department
University of Jaén
Linares (Jaén), Spain
nicolas@ujaen.es

***Abstract*—At the dawn of the fourth industrial revolution, artificial intelligence is impregnating our society by an overwhelming number of applications with rising complexity. This circumstance has triggered a new debate on explainable artificial intelligence in terms of transparency and confidence. Therefore, the relevance of interpretability is remarkable since this concept provides us with transparence and understandability in avoidance of black-box systems. On the other hand, cloud computing is a new paradigm of distributed computation based on the externalization of computing needs offered as services, whose performance has a significant economic and environmental impact and is strongly influenced by the scheduler, which is, likely, the most critical part in charge of allocation computational resources. In this regard, fuzzy rule-based systems are knowledge-based systems that are increasingly arising as an alternative for the development of cloud scheduling systems, mainly due to their intrinsic features such as adaptability to environments, dynamism and capability to cope with uncertainty. Bearing in mind the above-mentioned ideas, this paper presents a study-case in which the interpretability in fuzzy rule-based schedulers for cloud computing, obtained through automatic learning processed, has been analyzed. Results show how an inherent feature, like interpretability, vanishes due to the use of learning processes for both total execution time and power consumption optimization, which put the focus on the relevance of facing interpretability in this kind of systems.**

Keywords—green cloud computing, scheduling, fuzzy rule-based systems, interpretability, explainable artificial intelligence

I. Introduction

Fuzzy Rule-Based Systems (FRBS) have been adapted to a great range of fuzzy modeling, control and classification problems [1, 2]. Moreover, they have attracted the scheduling community attention for their application to large-scale scheduling [3, 4, 5]. A major advantage of FRBS is related to their ability to cope with noisy or uncertain information presented in highly dynamic systems. Therefore, the application of Fuzzy Systems as schedulers can be very beneficial to make scheduling decisions, for the improvement of the total execution time, following a human-like reasoning and thus to distribute workload effectively among the different resources [6, 7].

Nevertheless, the successful operation of FRBS is strongly related to the quality of their acquired knowledge or fuzzy Rule Bases (RB). Since obtaining such RBs on the basis of experts' criteria is not feasible in most of current applications, mainly due to the lack of experts in the specific area or because only a partial or incomplete description of the system can be derived, self-learning strategies have been pursued. In this sense, the role of genetic strategies has to be underlined. Genetic strategies have proved their effectiveness for the evolution of rules in FRBS based on the survival of best suited adapted chromosomes. Specifically, two main strategies, namely, Michigan [8] and Pittsburgh approaches [9] stand out of the genetic strategies which consider the encoding of rules and RB as individuals of the candidate population, respectively, and are subject to genetic operations.

In addition, new bio-inspired learning approaches for knowledge acquisition are recently emerging based on well-known optimization algorithms. This is the case of the adaptation Differential Evolution (DE) [10] which follows the general procedure of evolutionary algorithms and considers a weighted difference process among RBs to achieve optimization of rules. Additionally, other strategies based on Swarm Intelligence (SI) have been used as well. In this sense, Knowledge Acquisition with a Swarm Intelligence Approach (KASIA) [11, 12, 13] and Knowledge Acquisition with Rules as Particles (KARP) [5, 13] are worth mentioning. Thus, when using KASIA, knowledge bases are considered as particles, which is similar to Pittsburgh approach. However, when using KARP, rules are considered as particles, which is similar to Michigan Approach.

This work has been supported by action 1 of the research support plan of the University of Jaén.

978-1-7281-6932-3/20/$31.00 ©2020 IEEE

Hence, given the high dependence of the expert schedulers in cloud computing with the quality of the knowledge bases and thus, with the learning acquisition process, it can be relevant to analyze the performance of these strategies in terms of total execution time and computational effort.

In addition, currently, researchers are facing another important concern which is related to the sustainability of Clouds Data Centers (CDC). Certainly, CDC are increasingly being used by Information Technology (IT) Service Providers, such as Google, Amazon and Microsoft to meet the world's digital needs [14, 15]. CDC provides an efficient infrastructure to store a large amount of data along with all processing capabilities. Indeed, IT service providers run data centers 24/7 with thousands of servers and storage devices and network services to provide 99.99% availability of cloud services [16, 17, 18]. Thus, the current viability of a sustainable economic system is one of the main challenges facing the world, and the CDC represents one of the largest consumers of electricity. The CDC is estimated to consume more than 2.5% of electricity worldwide with a global economic impact of 30 billion dollars annually [19]. Additionally, it is estimated that data centers have been responsible for the emission of 2% of global pollutant emissions in recent decades [14, 20].

These figures encourage the application of innovative elements and disruptive measures in CDC to achieve greater efficiency in power consumption and reduce pollutant emissions. Among the possible areas of work in this regard, sustainable and green CDC require the application of multiple techniques and technologies. One of the techniques consists in optimizing the allocation of computational need to computational resources. In that sense, the schedulers are the software elements responsible for such assignment under the consideration of multiple criteria and configurations. They constitute one of the main components of the cloud processing structure that interact with other components such as the information systems of the resource management systems and the network maintenance systems to perform their function. In particular, it is necessary to design and to plan strategies capable of dealing with the inherent uncertainty and dynamism of cloud networks in order to reduce power consumption [21, 22] while optimizing the total execution time.

However, as far as authors are concerned, when facing optimized scheduling in terms of efficiency (total execution time) and sustainability (power consumption) with FRBS, there is an important lack in terms of interpretability. Were some efforts performed, in terms of interpretability, related to the efficiency and sustainability in CDC management, experts would be able to understand in a better way the internal relationships that lead to a clear improvement in scheduling processes. Therefore, this paper highlights the relevance of interpretability when designing fuzzy rule-base schedulers for cloud computing, and represents the initial efforts of authors in this issue.

The rest of the paper is organized as follows. Section II depicts some important aspects related to interpretability. Section III puts the focus on interpretability indexes. Section IV includes a study-case where interpretability is analyzed. Finally, conclusions and future actions are shown in section V.

II. Interpretability

Despite the absence of a clear definition for interpretability of knowledge based systems, there are several terms such as intelligibility and comprehensibility, with which interpretability could be interchangeable. It could be said that interpretability is the ability to understand the meaning of something, therefore, interpretability is always a desirable feature for every single kind of engineer solution. Furthermore, in the case of those applications involving knowledge based systems and due to the increasing importance of eXplainable Artificial Intelligence (XAI), it should be an essential requirement. Certainly, XAI is currently achieving relevance in order to provide easily comprehensible systems to users, so that people can feel confidence, and consequently, society can get benefit from a panoply of advantages due to the application of artificial intelligence. In this sense, besides accuracy, interpretability should be addressed when designing a knowledge based system. Nevertheless, it is important to point out that generating interpretable systems is not a straightforward task. Nowadays, an overwhelming number of fuzzy systems are obtained giving priority to accuracy, disregarding their interpretability, which could be considered an ill-decision because knowledge based systems could become black-boxes, despite its intrinsic and implicit interpretability possibilities.

In this point, it is important to highlight that, looking for a good interpretability-accuracy trade-off is one of the most complex tasks on system modeling, which demands the help of powerful software tools.

Fortunately, there are available open source software in form of libraries and tools that can be used for the development and modeling of systems taking into account not only the accuracy but also the interpretability. In this way, it is important to remark FrIDA [23], KEEL [24] and GUAJE [25, 26, 27], for instance.

FriDA (Free Intelligent Data Analysis Toolbox) is a Java-based graphical user interface in order to address a large number of data analysis programs [23]. Additionally, this toolbox is also equipped with basic visualization capabilities, like scatter plots, bar charts, and with specialized visualization modules for decision and regression trees as well as prototype-based classifiers. FriDA's architecture is formed by individual tools refer to the different data analysis methods. All parts of this toolbox (Java as well as C based) are free and open software under the Gnu Lesser (Library) Public License.

KEEL (Knowledge Extraction based on Evolutionary Learning) is a software that provides different tools in order to assess evolutionary algorithms for data mining problems including regression, classification, unsupervised learning, for instance [24]. Additionally, KEEL includes evolutionary learning algorithms based on different approaches: Pittsburgh, Michigan and Iterative Rule Learning (IRL), as well as the integration of evolutionary learning techniques with different pre-processing techniques, allowing it to perform a complete analysis of any learning model. It is important to point out that KEEL has been designed not only for educational but also for research purposes.

GUAJE (Generating Understandable and Accurate fuzzy models in a Java Environment) is a well-known open source and a dynamic user friendly tool whose main goal is the generation or adjustment of fuzzy knowledge bases in order to deal with specific problems. For this purpose, GUAJE implements the fuzzy modeling methodology called "Highly Interpretable Linguistic Knowledge [27, 28]. The whole process of using GUAJE is broken down in different, but not necessarily sequential, steps with the purpose of generating a knowledge base related to a specific problem, carefully integrating both expert and induced knowledge. GUAJE is a software that combines several preexisting software tools (not only libraries), with the purpose of building interpretable fuzzy models.

III. Interpretability Index considered

Besides the existence of software for the design of interpretable knowledge bases, the existence of methodologies to analyze the interpretability of FRBS are needed as well. In this regard, given a fuzzy rule-based system, there are two main kinds of approaches so as to analyze its interpretability [29]:

- Complexity-based Interpretability: These approaches are devoted to decreasing the complexity of the obtained model (usually measured as number of rules, variables, labels per rule, etc.).
- Semantics-based Interpretability: These approaches are devoted to preserve the semantics associated with the Membership Functions (MFs), redundancy, contradiction and inconsistency. We can find approaches trying to ensure semantic integrity by imposing constraints on the MFs or approaches considering measures such as distinguishability, coverage, etc.

In [29], authors propose a taxonomy by the combination of complexity-based interpretability and semantics-based interpretability, which leads to four quadrants shown in table I.

TABLE I. Taxonomy to Analyze the Interpretability of Linguistic Fuzzy Rule-Based Systems

	RULE BASE LEVEL	**FUZZY PARTITION LEVEL**
Complexity-based interpretability	***Q1*** • Number of rules • Number of conditions	***Q2*** • Number of membership functions • Number of features
Semantic-based interpretability	***Q3*** • Consistency of rules • Rules fires at the same time • Transparency of rule structure • Cointension	***Q4*** • Completeness of coverage • Normalization • distinguishability • complementarity • relative measures

Some works consider only one quadrant, while others may be related to several quadrants simultaneously. This could be explained because the improvement obtained in one quadrant could imply the improvement in other one (e.g. reducing the number of MFs (Q2) as a way to reduce the number of rules (Q1)). To be precise, some simple interpretability indexes, which consider the readability of the rule base, are usually used in multi-objective genetic fuzzy-based learning [30].

On the other hand, many interpretability indexes that can be found, like [31, 32, 33, 34, 35], put the focus on the readability of fuzzy partitions. This interpretability indexes are based on the evaluation of main partition properties like distinguishability, coverage, overlapping, and similarity. These interpretability indexes are usually used in order to preserve the readability of fuzzy rule-based systems automatically generated, and in order to increase the accuracy by means of tuning processes.

In [36], the author proposes a numerical index in order to assess the fuzzy rule-based classification systems. This index considers three different terms related to the number of premises, number of labels and coverage. In addition, in [37], authors provide several general interpretability indexes to be applied to any scatter or linguistic model implemented by any type of membership functions.

Finally, this work considers the interpretability index suggested in [28, 38], which is a fuzzy interpretability index obtained from a hierarchical fuzzy system. Input and output variables are shown in table II, where N is the minimum number of rules.

TABLE II. Input and Output Variables

VARIABLE	**UNIVERSE OF DISCOURSE**	**NUMBER OF LABELS**
Total number of rules	[N,8 x N]	3
Total number of premises	[N,16 x N]	2
Number of rules with one input	[0,8 x N]	2
Number of rules with two inputs	[0,8 x N]	2
Number of rules with three or more inputs	[0,8 x N]	2
Average number of labels by input	[2,9]	2
Rule base dimension	[0,1]	3
Rule base complexity	[0, 1]	3
Rule base Interpretability	[0, 1]	5
Interpretability index	[0, 1]	5

These variables are taken as inputs of a hierarchical fuzzy system and they are grouped according to the information they convey. Therefore, the Interpretability Index is computed as the result of inference of a hierarchical fuzzy system that is broken down in four linked KBs, see Fig. 1. A first rule base, called "Rule Base Dimension", makes an estimation of the rule base dimension taking into account as inputs the total number of rules and premises. At the same time, a second rule base, called "Rule Base Complexity," assesses the rule base complexity bearing in mind the number of inputs used by the rules (one input, two inputs three or more inputs). Additionally, a third rule base, called "Rule Base Interpretability", combines the rule base dimension and the complexity, through its outputs, and it yields a rule base interpretability index.

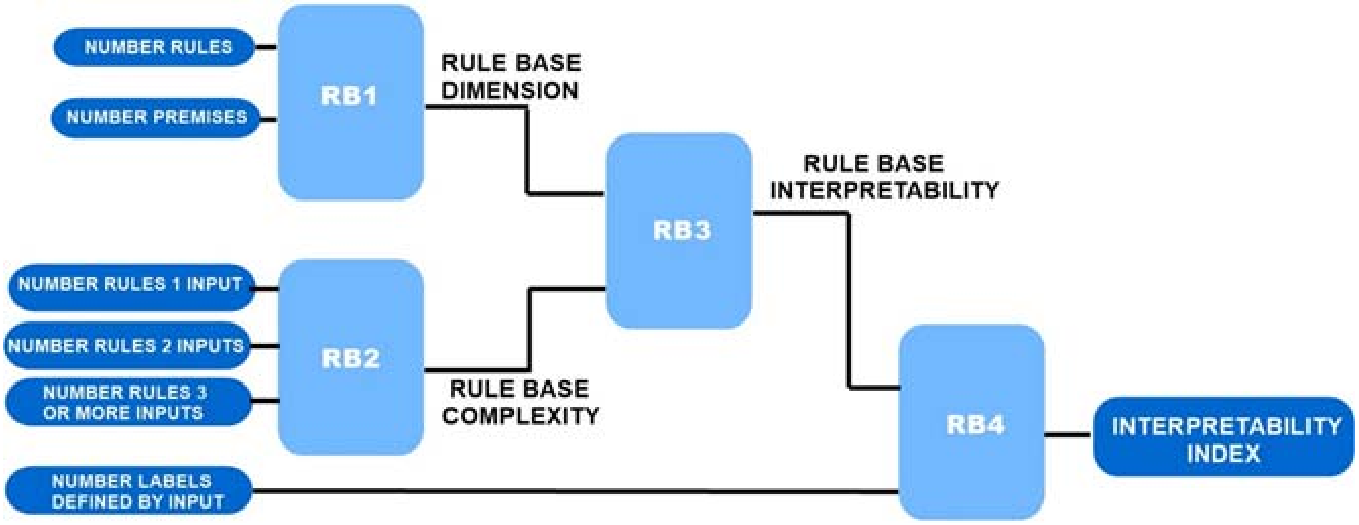

Fig. 1. Hierarchical fuzzy system for assessing interpretability.

Finally, a rule base integrates the rule base interpretability with the evaluation of interpretability for the system variables, considering the total number of labels per input and assuming that the fuzzy rule-based systems to be evaluated only include strong fuzzy partitions, which means that these partitions satisfy the followings conditions:

$$\forall\, x \in \cup, \textstyle\sum_{i=1}^{N} \mu_{A_i}(x) = 1 \quad (1)$$

$$\forall\, A_i, \exists\, x \in \cup \,/\, \mu_{A_i}(x) = 1 \quad (2)$$

Where U is the Universe of discourse, N is the number of labels and $\mu_{A_i}(x)$ is the membership degree of x to the A_i fuzzy set.

It is important to point out that each end every rule base has been implemented in the form of Mamdani rules considering product t-norm as conjunctive operator, sum t-conorm for aggregation, and the winner rule fuzzy reasoning mechanism.

The different rule bases used by the hierarchical fuzzy system for the assessment interpretability in FRBS can be found and explained in detail in [28, 38].

This paper proposes the use of this hierarchical fuzzy system for the interpretability analysis of the different fuzzy rule-based schedulers for the optimization of the total execution time and the power consumption.

IV. Study-Case

In order to analyze the interpretability in fuzzy rule-based schedulers in cloud computing, several experiments have been carried out. To be precise, two different learning processes have been ran so as to achieve two different schedulers. The first one is devoted to optimize the cloud system performance in terms of total execution time, while the second one is aimed to optimize the power consumption. It is important to highlight that both schedulers have been obtained by the use of a genetic fuzzy systems with Pittsburgh approach. In order to obtain every single fuzzy rule-based scheduler, 30 experiments have been carried out with a population of 20 knowledge bases, selection rate of 90%, and mutation rate of 10%, which decreases with generations.

Experiments that have been carried out consider a cloud system topology involving a cloud data center with 20 worker hosts, which are running 5 Virtual Machines (VM). Every single VM has an image size of 1000MB, 512MB Random Access Memory (RAM), 2000 millions of instructions per second (MIPS) and 2 CPUs. In this scenario, the Montage workflow for mosaic images composition [39] has been executed. Specifically, results are presented for a Montage data trace with synthetic 1000 Node Digital Asset eXchange (DAX) workflows provided in [40], which uses XML files following a specific format, Directed Cyclic Graph, leading to DAX files. These synthetic workflows real applications.

To run the simulations WorkflowSim simulator [41], which was modified in [42] to implement power simulation capabilities, has been used. In addition, the performance of five traditional workflow scheduling strategies is evaluated with the same Montage workflows to obtain compared results:

- STATIC (Static algorithm). Schedule based on workflow planner to set the mapping relationship.
- FCFS (First Come First Served): It designates each job in the arriving order to the next available VM host, disregarding jobs' expected completion time on that VM host. If there are diverse resources available, it selects a resource in a random way.
- MCT (Minimum Completion Time): MCT allocates each workflow job arbitrarily to the available VM host with the best supposed completion time for the job.
- MINMIN (MinMin): The MinMin scheduling strategy orders the workflow jobs in order of completion time. The workflow job with the minimum completion time is chosen

and associated to the VM host with the minimum completion time expectation. Next, the new incoming jobs are considered to order jobs and the process is repeated meanwhile there exist non-scheduled jobs. The objective of the MinMin strategy is to minimize the overall execution time by the prioritizing of the shortest jobs.

- MAXMIN (MaxMin): The MaxMin strategy is similar to the MinMin strategy, however, the MaxMin strategy selects jobs with the maximum completion time and designates it to the VM host with minimum completion time. The goal of MaxMin is to reduce cost of the more computing demanding jobs.

A. *Obtained Fuzzy Scheduler for Time Optimization (SCH-1)*

This Scheduler is aimed to optimize the performance of cloud systems in terms of total execution time. To be precise, the fitness used to evaluate the performance during the learning process is given by the *makespan*, which is defined by

$$makespan = max_{j \in J} T_j \tag{3}$$

Where T_j is the finish time of job j, included in the complete set of jobs J of the workflow. The rule base is shown in table III, while its variables, which entail uniform fuzzy partitions, are presented in table IV and its membership functions depicted in fig. 7a).

TABLE III. OBTAINED RULES FOR TIME OPTIMIZATION

RULE BASE (SCH-1)
IF PEs is low and BW is not low and RAM is not medium and TRANSFERT is not medium **THEN** SELECTION is high
IF MIPS is not medium and PEs is high and BW is not low and RAM is not low and TRANSFERT is medium **THEN** SELECTION is high
IF BW is medium and RAM is medium and TRANSFERT is low **THEN** SELECTION is high
IF MIPS is not medium and PEs is low and BW is not low and TRANSFERT is low **THEN** SELECTION is low
IF MIPS is low and PEs is not low and BW is not low and TRANSFERT is not medium **THEN** SELECTION is medium
IF MIPS is medium and PEs is not medium and BW is low and RAM is low and TRANSFERT is not low **THEN** SELECTION is low
IF MIPS is not medium or BW is low or RAM is low or TRANSFERT is high **THEN** SELECTION is low
IF MIPS is not high or PEs is low or BW is medium or RAM is medium **THEN** SELECTION is very-high
IF MIPS is medium or PEs is high or RAM is low or TRANSFERT is medium **THEN** SELECTION is high
IF MIPS is high and PEs is medium and BW is not low and RAM is not low and TRANSFERT is not medium **THEN** SELECTION is high

TABLE IV. INPUT AND OUTPUT VARIABLES (SCH-1)

SCHEDULER FOR TIME OPTIMIZATON			
Input Variables	***Description***	***Universe of Discourse***	***Number of labels***
MIPS	Millions Instructions per Second being used in host VM	[0-40,000]	3
PE	Processing elements being used in host VM	[0-300]	3
BW	Bandwidth being used by in host VM	[0-15,000]	3
RAM	RAM being used in host VM	[0-8000]	3
TRANSFERT	Transfer time for host VM	[040,000,000]	3
Output Variable	***Description***	***Universe of Discourse***	***Number of labels***
SELECTION	Suitability of every single host VM	[0-1]	5

Finally, the obtained accuracy in terms of total execution time is shown in table VII, which additionally portrays a comparison to traditional schedulers.

B. *Obtained fuzzy scheduler for the optimization of power consumption (SCH-2)*

This Scheduler is aimed to optimize the performance of cloud systems in terms of power consumption. The rule base is shown in table V, while its variables, which entail uniform fuzzy partitions, are presented in table VI and its membership functions depicted in fig. 7b).

TABLE V. OBTAINNED RULES FOR POWER CONSUMPTION OPTIMIZATION

RULE BASE (SCH-2)
If MIPS is medium or POWER is high or LENGTH is not medium or POWERMAX is medium or USE is not low **THEN** SELECTION is low
If MIPS is medium and POWER is high and LENGTH is high and POWERMAX is medium and USE is not low **THEN** SELECTION is not medium
If MIPS is medium or POWER is low or LENGTH is high or POWERMAX is medium or USE is not low **THEN** SELECTION is very-low
If MIPS is low and POWER is high and LENGTH is medium and POWERMAX is low and USE is high **THEN** SELECTION is not very-low
If MIPS is low or POWER is medium or LENGTH is medium or POWERMAX is medium or USE is not low **THEN** SELECTION is very-high
IF MIPS is not low or POWER is high or LENGTH is high or POWERMAX is medium or USE is medium **THEN** SELECTION is not very-low
If MIPS is high and POWER is high and LENGTH is low and POWERMAX is not medium and USE is medium **THEN** SELECTION is not high
If MIPS is not high and POWER is low and LENGTH is not high and POWERMAX is low **THEN** SELECTION is not very-low
If MIPS is medium or POWER is low or LENGTH is high or POWERMAX is high or USE is high **THEN** SELECTION is medium
If MIPS is not high or POWER is not medium or LENGTH is medium or POWERMAX is low or USE is high **THEN** SELECTION is medium
If MIPS is medium or POWER is high or LENGTH is medium or POWERMAX is low or USE is high **THEN** SELECTION is very-high
If MIPS is low or POWER is high or LENGTH is medium or POWERMAX is low or USE is medium **THEN** SELECTION is medium
If MIPS is medium or LENGTH is not medium or POWERMAX is low or USE is medium **THEN** SELECTION is very-high
If MIPS is medium or POWER is high or LENGTH is low or POWERMAX is high or USE is not low **THEN** SELECTION is high
If MIPS is low and LENGTH is high and POWERMAX is high and USE is low **THEN** SELECTION is medium
If MIPS is low and POWER is high and LENGTH is not low and POWERMAX is medium and USE is low **THEN** SELECTION is very-high
If MIPS is medium or POWER is low or LENGTH is low or USE is low **THEN** SELECTION is not high

TABLE VI. INPUT AND OUTPUT VARIABLES (SCH-1)

SCHEDULER FOR TIME OPTIMIZATON			
Input Variables	***Description***	***Universe of Discourse***	***Number of labels***
MIPS	Millions Instructions per Second	[0-40,000]	3
POWER	Idle Power Consumption	[33-260]	3
LENGTH	Task Size	[110-89120]	3
POWERMAX	Dynamic Power Consumption	[47-372]	3
USE	Utilization factor of the Process Entity	[0-1]	3
Output Variable	***Description***	***Universe of Discourse***	***Number of labels***
SELECTION	Suitability of every single host VM	[0-1]	5

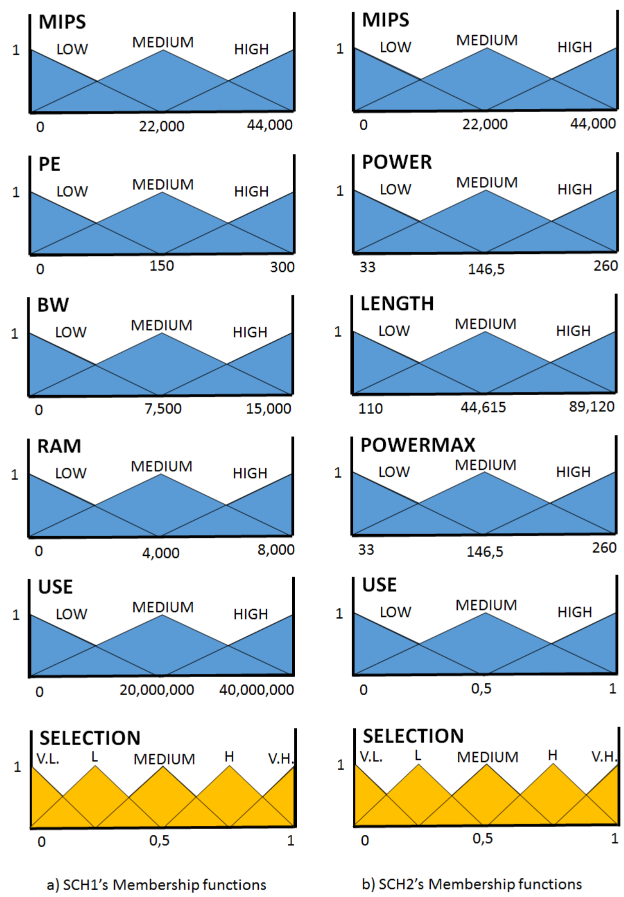

Fig. 7. Membership functions used in schedulers.

Finally, the obtained accuracy in terms of power consumption is shown in table VII, which additionally portrays a comparison to traditional schedulers.

TABLE VII. ACCURACY OF SCHEDULERS FOR BOTH TOTAL EXECITION TIME AND POWER CONSUMPTION OPTIMIZATION

ACCURACY OBTAINED BY SCHEDULERS		
Name	***Total Execution Time (s)***	***Power Consumption (W)***
Static	5.6930e+03	22.884907e+6
FCFS	1.2882e+03	1548344e+6
Min-Min	1.2887e+03	1292709e+6
Max-Min	1.2885e+03	1290256e+6
MCT	1.2882e+03	1287874e+6
Proposal	**1.0899e+03**	**1285562e+6**

C. Interpretability analysis

Once these fuzzy rule-based schedulers have been obtained, an interpretability analysis has been carried out. The results of this analysis shows how its interpretability are almost inexistent in both schedulers. In fact, the interpretability index for SCH-1 is 0.2952, while the interpretability index for SCH-2 is 0.2867. In this regard, and bearing in mind the knowledge bases involved in the estimation of the interpretability index, in both cases (SCH1 and SCH-2), the dimension, due to the number of premises, and complexity are very high with the exception of SCH-1, whose complexity is medium. Anyway, the average number of labels used undermines the interpretability index due to the number of premises. In addition, according to authors opinion, bearing in mind their experience on the field, the interpretability of the obtained knowledge bases is not as good as could be desired. In fact, authors find several problems in order to understand the obtained knowledge bases, specially due to the number of rules and to their combination, to the number of premises and the two link operators used ("or" and "and"). Moreover, authors consider that despite the fact the obtained indexes could be a nice indicator on the interpretability, the second one should be lower, in comparison terms, because of its high number of rules.

The obtained interpretability indexes could be considered as clearly improvable, so that these results indicate that additional efforts should be carried out when designing and developing fuzzy rule-based schedulers for cloud computing, not only when optimizing total execution time, but also when optimizing power consumption, or even both in a simultaneous way by means of a multi-objective approach. In this regard, the room for improvement depicted in this work allows the acquisition of interpretable fuzzy rule-based schedulers for cloud computing, whose accuracy will not be negatively influenced when considering interpretability in the learning process.

V. CONCLUSIONS AND FUTURE ACTIONS

This work has analyzed the interpretability in two different fuzzy rule-based schedulers for both total execution time and power consumption optimization, respectively. Once it has been corroborated that the intrinsic interpretability feature of FRBS vanishes when obtaining fuzzy rules-based schedulers throughout the use of a learning process, this paper encourages to use interpretability indexes in order to obtain efficient and sustainable schedulers in cloud computing in such a way that the improvement obtained in interpretability do not undermines the obtained accuracy in terms of total execution time and of power consumption. In this regard, taking into account the above mentioned ideas and the obtained results, the authors are going to explore three different multi-objective approaches in order to design fuzzy rule-based schedulers for cloud computing regarding the goal to optimize: (1) optimization of total execution time and interpretability, (2) optimization of power consumption and interpretability, and (3) optimization of total execution time and of power consumption, considering interpretability when selecting non-dominated solutions in the Pareto front. In this sense, it is important to highlight the first two approaches entail a trade-off between accuracy and interpretability. Nevertheless, the last one considers interpretability without assuming a cost in terms of accuracy, neither in total execution time nor in power consumption.

ACKNOWLEDGMENT

This work has been supported by action 1 of the research support plan of the University of Jaén.

REFERENCES

[1] Nikolaos L. Tsakiridis, John B. Theocharis, Panos Panagos, George C. Zalidis, An evolutionary fuzzy rule-based system applied to the prediction of soil organic carbon from soil spectral libraries, Applied Soft Computing, Volume 81, 2019. 105504.

[2] J.M. Soto-Hidalgo A.Vitiello, J.M. Alonso, G. Acampora, J. Alcala-Fdez. Design of Fuzzy Controllers for Embedded Systems With JFML. International Journal of Computational Intelligence Systems 12(1). 2019, 204–214.

[3] Carsten Franke, Frank Hoffmann, Joachim Lepping, Uwe Schwiegelshohn, Development of scheduling strategies with Genetic Fuzzy systems, Applied Soft Computing, Volume 8, Issue 1, 2008, Pages 706-721.

[4] Amit K. Shukla, Rahul Nath, Pranab K. Muhuri, Q.M. Danish Lohani, Energy efficient multi-objective scheduling of tasks with interval type-2 fuzzy timing constraints in an Industry 4.0 ecosystem, Engineering Applications of Artificial Intelligence, Volume 87, 2020, 103257.

[5] S. García-Galán, R.P. Prado, J.E. Muñoz Expósito, Rules discovery in fuzzy classifier systems with PSO for scheduling in grid computational infrastructures, Applied Soft Computing, Volume 29, 2015, Pages 424-435.

[6] R. P. Prado, S. García-Galán, A. J. Yuste, and J. E. Muñoz Expósito. Genetic fuzzy rule-based scheduling system for grid computing in virtual organizations. *Soft Comput.*, vol. 15, no. 7. pp. 1255–1271, 2011.

[7] R. P. Prado, S. García-Galán, A. J. Yuste, and J. E. Muñoz Expósito. A fuzzy rule-based meta-scheduler with evolutionary learning for grid computing. *Eng. Applicat. Artif. Intell.*, vol. 23, no. 7, pp. 1072–1082, 2010.

[8] L.B. Booker, D.E. Goldberg, J.H. Holland.Classifier systems and genetic algorithms. Artificial Intelligence, Volume 40, Issues 1–3, 1989, Pages 235-282.

[9] S. F. Smith. A Learning systems based on genetic adaptive algorithm. PhD. Thesis, Pitssburgh, PA, USA, 1980.

[10] R. P. Prado, S. García-Galán, J. E. Muñoz-Expósito. Kasia Approach vs. Differential Evolution in Fuzzy Rule-Based Meta-Schedulers for Grid Computing. IEEE 5th International Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS 1011). Paris 2011.

[11] R. P. Prado, S. García-Galán, J. E. M. Exposito and A. J. Yuste, "Knowledge Acquisition in Fuzzy-Rule-Based Systems With Particle-Swarm Optimization," in *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 6, pp. 1083-1097, Dec. 2010.

[12] S. García-Galán, R.P. Prado, J.E. Muñoz Expósito. Fuzzy scheduling with swarm intelligence-based knowledge acquisition for grid computing, Engineering Applications of Artificial Intelligence. Volume 25, Issue 2, 2012, Pages 359-375.

[13] S. García-Galán, R. P. Prado and J. E. M. Expósito, "Swarm Fuzzy Systems: Knowledge Acquisition in Fuzzy Systems and Its Applications in Grid Computing," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 7, pp. 1791-1804, July 2014.

[14] Junaid Shuja, Abdullah Gani, Shahaboddin Shamshirband, Raja Wasim Ahmad, Kashif Bilal. Sustainable Cloud Data Centers: A survey of enabling techniques and technologies. Renewable and Sustainable Energy Reviews. Volume 62, 2016, Pages 195-214.

[15] Srinivasan S. Cloud Computing Evolution. In: Cloud Computing Basics. SpringerBriefs in Electrical and Computer Engineering. (2014). Springer, New York, NY.pp 1-16.

[16] A. Ali, Li Lu, Y. Zhu, and J. Yu. An Energy Efficient Algorithm for Virtual. Machine Allocation in Cloud Datacenters. CCIS 626, pp. 61–72, 2016.

[17] K. Bilal, S.U.R. Malik, O. Khalid, A. Hameed, E. Alvarez, V. Wijaysekara et al.A taxonomy and survey on green data center networks. Future Genereration Computer Systems, vol.36, pp. 189–208, 2014.

[18] J. Shuja, S. Madani, K. Bilal, K. Hayat, S. Khan, S. Sarwar. Energy-efficient data centers. Computing, vol. 94, pp. 973–94, 2012.

[19] G. Meijer. Cooling energy-hungry data centers. Science; vol. 328(5976), pp. 318–9, 2010.

[20] Brown R. Report to congress on server and data center energy efficiency public law 109-431. Environ. Prot, pp. 109:431, 2007.

[21] A. Beloglazov, J. Abawajy, R. Buyya. Energy-aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing. Future Gener Comput Systems, vol. 28(5), pp. 755–768, 2012.

[22] A. Bala and I. Chana. Autonomic fault tolerant scheduling approach for scientific workflows in Cloud computing. Concurrent Engineering: Research and Applications, vol. 23(1), pp. 27–39, 2015.

[23] C. Borgelt and G. González-Rodríguez. FrIDA- a free intelligent data analysis toolbox. In IEEE International Conference on Fuzzy Systems, pages 1892–1896, 2007.

[24] J. Alcalá-Fdez et al. KEEL: A software tool to assess evolutionary algorithms for data mining problems. Soft Computing, 13(3):307–318, 2009-

[25] D. P. Pancho, J. M. Alonso, and L. Magdalena, Quest for interpretability-accuracy trade-off supported by Fingrams into the fuzzy modeling tool GUAJE, International Journal of Computational Intelligence Systems, 6(1):46-60, 2013.

[26] J. M. Alonso and L. Magdalena, Generating understandable and accurate fuzzy rule-based systems in a java environment, Lecture Notes in Artificial Intelligence - 9th International Workshop on Fuzzy Logic and Applications, LNAI6857, 212-219, 2011.

[27] J. M. Alonso and L. Magdalena, HILK++: an interpretability-guided fuzzy modeling methodology for learning readable and comprehensible fuzzy rule-based classifiers, Soft Computing, 15(10):1959-1980, 2011.

[28] J. M. Alonso, L. Magdalena and S. Guillaume, HILK: a new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism, International Journal of Intelligent Systems, 23(7):761-794, 2008.

[29] M. J. Gacto, R. Alcala, F. Herrera, "Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures", *Inf. Sci.*, vol. 181, no. 20, pp. 4340-4360, 2011.

[30] H. Ishibuchi, Y. Nojima, Analysis of interpretability–accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning, International Journal of Approximate Reasoning 44 (2007) 4–31.

[31] A. Botta, B. Lazzerini, F. Marcelloni, D.C. Stefanescu, Context adaptation of fuzzy systems through a multi-objective evolutionary approach based on a novel interpretability index, Soft Computing 13 (5) (2009) 437–449.

[32] P. Fazendeiro, J.V. de Oliveira, A working hypothesis on the semantics/accuracy synergy, in: Joint EUSFLAT-LFA 2005, Barcelona, Spain, September 7–9, 2005, pp. 266–271.

[33] Y. Jin, W. von Seelen, B. Sendhoff, On generating FC3 fuzzy rule systems from data using evolution strategies, IEEE Transactions on Systems, Man, and Cybernetics 29 (6) (1999) 829–845.

[34] C. Mencar, Distinguishability quantification of fuzzy sets, Information Sciences 177 (1) (2007) 130–149.

[35] M. Setnes, R. Babuska, U. Kaymak, H.R. van Nauta Lemke, Similarity measures in fuzzy rule base simplification, IEEE Transactions on Systems, Man, and Cybernetics, Part B 28 (3) (1998) 376–386.

[36] D. Nauck, Measuring interpretability in rule-based classification systems, in: FUZZ-IEEE 2003, vol. 1, St. Louis, Missouri, USA, May 25–28, 2003, pp. 196–201.

[37] M. Galende, J.M Gacto, G Sainz, R. Alcalá. Comparison and Design of Interpretable Linguistic vs. Scatter FRBSs: GM3M Generalization and New Rule Meaning Index (RMI) for Global Assessment and Local Pseudo-Linguistic Representation. Information Sciences 282 (2014) 190-213.

[38] J.M. Alonso, S. Guillaume, L. Magdalena, A hierarchical fuzzy system for assessing interpretability of linguistic knowledge bases in classification problems, in: IPMU 2006, Information Processing and Management of Uncertainty in Knowledge-Based Systems, Paris, France, July 2–7, 2006, pp. 348– 355.

[39] "Montage Project". http://montage.ipac.caltech.edu.

[40] Pegasus, https://confluence.pegasus.isi.edu/display/pegasus/workflowgenerator

[41] W. Chen and E. Deelman, "WorkflowSim: A toolkit for simulating scientific workflows in distributed environments," in *2012 IEEE 8th International Conference on E-Science (e-Science)*, Chicago, IL, 2012 pp. 1-8.

[42] I.T. Cotes-Ruiz, R.P. Prado, S. García-Galán, J.E. Muñoz-Expósito, N. Ruiz-Reyes. Dynamic Voltage Frequency Scaling Simulator for Real Workflows Energy-Aware Management in Green Cloud Computing Plos One, vol. 12, no. 1, 2017